Cisco ASR 1000 Series Aggregation Services Routers Software Configuration Guide

May 22, 2015

Cisco Systems, Inc.
www.cisco.com

Cisco has more than 200 offices worldwide. Addresses, phone numbers, and fax numbers are listed on the Cisco website at www.cisco.com/go/offices.

Text Part Number: OL-16506-15

THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS MANUAL ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS MANUAL ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. USERS MUST TAKE FULL RESPONSIBILITY FOR THEIR APPLICATION OF ANY PRODUCTS.

THE SOFTWARE LICENSE AND LIMITED WARRANTY FOR THE ACCOMPANYING PRODUCT ARE SET FORTH IN THE INFORMATION PACKET THAT SHIPPED WITH THE PRODUCT AND ARE INCORPORATED HEREIN BY THIS REFERENCE. IF YOU ARE UNABLE TO LOCATE THE SOFTWARE LICENSE OR LIMITED WARRANTY, CONTACT YOUR CISCO REPRESENTATIVE FOR A COPY.

The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB's public domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

NOTWITHSTANDING ANY OTHER WARRANTY HEREIN, ALL DOCUMENT FILES AND SOFTWARE OF THESE SUPPLIERS ARE PROVIDED "AS IS" WITH ALL FAULTS. CISCO AND THE ABOVE-NAMED SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THOSE OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE.

IN NO EVENT SHALL CISCO OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THIS MANUAL, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: www.cisco.com/go/trademarks. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1110R)

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.

Cisco ASR 1000 Series Aggregation Services Routers Software Configuration Guide
Copyright © 2008 - 2015Cisco Systems, Inc. All rights reserved.

Preface

Revised: December 12, 2014, OL-16506-17

This preface describes the objectives and organization of this document and explains how to find additional information on related products and services. This preface contains the following sections:

- Objectives, page 1
- Document Revision History, page 2
- Organization, page 4
- Related Documentation, page 6
- Conventions, page 6
- Obtaining Documentation and Submitting a Service Request, page 7

Objectives

This document provides an overview of software functionality that is specific to the Cisco ASR 1000 Series Aggregation Services Routers. It is not intended as a comprehensive guide to all of the software features that can be run using the Cisco ASR 1000 Series Routers, but only the software aspects that are specific to these routers.

For information on general software features that are also available on the Cisco ASR 1000 Series Routers, see the Cisco IOS XE technology guide for that specific software feature.

Document Revision History

The Document Revision History records technical changes to this document. The table shows the Cisco IOS XE software release number and document revision number for the change, the date of the change, and a brief summary of the change.

Release No.	Date	Change Summary
IOS XE 3.13.1	December 12, 2014	• Added "PPP Half-Bridge on the Cisco ASR 1000 Series Routers" chapter.
IOS XE 3.13	July 30, 2014	• Added new content to document IEEE 1588v2 PTP Support in the "IEEE 1588v2 PTP Support" chapter. • Added HDLC-Ethernet Interworking feature updates in the "High-Level Data Link Control-Ethernet Interworking, page 39" section of the "Configuring MPLS Layer 2 VPNs"
IOS XE 3.12	March 28, 2014	• Added Multi Member-link MLPPPoA or MLPPPoEoA feature updates in the "Cisco IOS XE Scaling Limits for MLP Bundles" section of the "Multilink PPP Support for the Cisco ASR 1000 Series Routers" chapter.
IOS XE 3.12	March 28, 2014	• Added Minimal Disruptive Restart Phase 3 feature updates in the "Software Upgrade Processes Supported by Cisco ASR 1000 Series Routers" chapter.
IOS XE 3.12	March 28, 2014	• Added MVPNv6 support for the "MVPN MLDP over GRE" feature in the "LSM-MLDP-based MVPN Support" chapter.
IOS XE 3.11.0	November 21, 2013	• Added MVPNv4 support for the "MVPN MLDP over GRE" feature in the "LSM-MLDP-based MVPN Support" chapter.
IOS XE 3.10.0	July 30, 2013	• Added "Packet Trace"chapter.
IOS XE 3.9.0	March 28, 2013	• Added new content to document UniDirectional Link Detection (UDLD) Protocol in the "UniDirectional Link Detection (UDLD) Protocol" chapter.
IOS XE 3.8.0	November 28, 2012	• Added Minimal Disruptive Restart Process section in the "Software Upgrade Processes Supported by Cisco ASR 1000 Series Routers" chapter.
IOS XE 3.7.0	July 25, 2012	• Added new content to document the Configuring Cisco Right-To-Use License in the "Configuring a Cisco Right-To-Use License" chapter. • Added new content to the Synchronous Ethernet in the "Network Synchronization Support" chapter. • Added new content to document the Walk-by User Support for PWLAN in ISG in the "Broadband Scalability and Performance" chapter. • Added new content to the bridge domain interface configurations in the "Configuring Bridge Domain Interfaces" chapter.

IOS XE 3.6.0	March 30, 2012	• Added new content to document the Cisco License Call Home in the "Configuring Cisco License Call Home" chapter. • Added new content to document the Frame Relay to ATM Bridged Interworking in the "Configuring MPLS Layer 2 VPNs" chapter. • Added new content to document the xconnect support on GEC (VPWS) on ASR1000 in the "Configuring MPLS Layer 2 VPNs" chapter.
IOS XE 3.4.0S	July 25, 2011	• Added new content to document the Monitoring and Maintaining Multilink Frame Relay feature in the "Monitoring and Maintaining Multilink Frame Relay" chapter. • Added new content to document In Service One-Shot Software Upgrade Procedure in the "Software Upgrade Processes Supported by Cisco ASR 1000 Series Routers" chapter.
IOS XE 3.3.0S	March 30, 2011	• Added new content to document the Scalability and Performance in the "Broadband Scalability and Performance" chapter. • Added software upgrade information for ASR 1001 chassis in the "Using Subpackages for Software Upgrade on a Cisco ASR 1001 Router or a Cisco ASR 1002-X Router" section in "Software Upgrade Processes Supported by Cisco ASR 1000 Series Routers" chapter.
IOS XE 3.2.0S	November 24, 2010	• Added new content to document bridge domain interface configurations in the "Configuring Bridge Domain Interfaces" chapter. • Added new content to document the Synchronous Ethernet in the "Network Synchronization Support" chapter.
IOS XE 3.1.0S	July 30, 2010	• Added upgrade and file package information in the "Software Packaging and Architecture" chapter. • Added restriction regarding ISSU support in Cisco IOS XE Release 3.1S in "Software Upgrade Processes Supported by Cisco ASR 1000 Series Routers" chapter.
IOS XE 2.6	February 26, 2010	• Added support for the Call Home feature in the "Configuring Call Home" chapter. • Moved the ISSU compatibility tables into the *Release Notes for Cisco ASR 1000 Series Aggregation Services Routers* document. • Relocated other ISSU compatibility information into the "Software Upgrade Processes Supported by Cisco ASR 1000 Series Routers" chapter.

IOS XE 2.5	December 15, 2009	Updated the "Software Upgrade Processes Supported by Cisco ASR 1000 Series Routers" chapter with new compatibility information.
IOS XE 2.4	June 30, 2009	• Updated the "Software Upgrade Processes Supported by Cisco ASR 1000 Series Routers" chapter with new compatibility information. • Updated book with installation information for the new optional subpackage for the Cisco WebEx Node for ASR 1000 Series.
IOS XE 2.3	February 27, 2009	Updated the "Software Upgrade Processes Supported by Cisco ASR 1000 Series Routers" chapter with new compatibility information.
IOS XE 2.2	November 20, 2008	Updated the "Software Upgrade Processes Supported by Cisco ASR 1000 Series Routers" chapter with new compatibility information.
IOS XE 2.2	November 3, 2008	Added new content to document compatibility of different versions of Cisco IOS XE software in the "Software Upgrade Processes Supported by Cisco ASR 1000 Series Routers" chapter.
IOS XE 2.2	October 3, 2008	Updated book to document ISSU procedures for Cisco IOS XE Release 2.2. Removed ISSU procedures from the "Consolidated Packages and SubPackage Management" chapter of the book and created the "Software Upgrade Processes Supported by Cisco ASR 1000 Series Routers" chapter.
IOS XE 2.2	September 23, 2008	Added the "Troubleshooting Software Mismatch with ESP Board ASR1000-ESP10-N" section for the Cisco ASR 1000 Embedded Services Processor 10G Non Crypto Capable feature.
IOS XE 2.1	July 2, 2008	Added the "Configuring and Accessing the Web User Interface" section. Added Appendix "Configuration Examples".
IOS XE 2.1	May 2, 2008	First release of the book.

Organization

This document contains the following chapters:

Title	Description
Software Packaging and Architecture	Provides an introduction to Cisco ASR 1000 Series Routers software packaging, processes, and file systems.
Using Cisco IOS XE Software	Provides an introduction to accessing the command-line interface (CLI) and using the Cisco software and related tools.

Title	Description
Console Port, Telnet, and SSH Handling	Provides an overview and configuration options for the handling of incoming console port, telnet, and SSH traffic on the Cisco ASR 1000 Series Routers.
Consolidated Packages and SubPackage Management	Provides information on downloading, installing, and running the software on the Cisco ASR 1000 Series Routers.
Software Upgrade Process	Provides information on ISSU compatibility and performing limited-downtime software upgrades on the Cisco ASR 1000 Series Routers.
High Availability Overview	Provides an overview of the High Availability architecture, behavior, and features on the Cisco ASR 1000 Series Routers.
Scalability and Performance	Provides information on scaling and performance on the Cisco ASR 1000 Series Routers.
Cisco License Call Home	Provides information on activating the Cisco License for Call Home feature.
Configuring Call Home for Cisco ASR 1000 Series Aggregation Services Routers	Provides information about the Call Home feature, which supports e-mail-based and web-based notification of critical system events. A versatile range of message formats are available for optimal compatibility with pager services, standard e-mail, or XML-based automated parsing applications.
Configuring Cisco Right-To-Use License	Provides information on activating the Cisco Right-To-Use License.
UniDirectional Link Detection (UDLD) Protocol	Provides an overview and configuration options for the UDLD protocol on the Cisco ASR 1000 Series Routers.
Using the Management Ethernet Interface	Provides an overview and configuration options for the Management Ethernet interface on the Cisco ASR 1000 Series Routers.
Synchronous Ethernet Support On The Cisco ASR 1000 Series Routers	Provides an overview and configuration options for the Synchronous Ethernet on the Cisco ASR 1000 Series Routers.
Configuring Ethernet Over Soft GRE	Provides an overview and configuration for the Generic Routing Encapsulation (GRE) is a tunneling protocol that can encapsulate a wide variety of network layer protocols inside virtual point-to-point links over an IP internetwork.
Configuring Bridge Domain Interfaces	Provides an overview of configuration options for the Bridge Domain Interface on the Cisco ASR 1000 Series Routers.
Monitoring and Maintaining Multilink Frame Relay	Provides an overview of monitoring and maintaining the Multilink Frame Relay feature for the Cisco ASR 1000 Series Routers.
Configuring MPLS Layer 2 VPNs	Provides an overview of the Frame Relay to ATM Bridged Interworking feature.

Title	Description
LSM-MLDP-based MVPN Support	Provides information on the Label Switched Multicast (LSM) feature supports IPv4 and IPv6 multicast traffic over a Multi-Protocol Label Switching (MPLS) network.
Tracing and Trace Management	Provides an overview of tracing on the Cisco ASR 1000 Series Router, and how to manage the tracing process and files.
PPP Half-Bridge on the Cisco ASR 1000 Series Routers	Provides information about PPP half-bridge on the Cisco ASR 1000 Series Routers.
Configuring and Accessing the Web User Interface	Provides an overview of the Cisco ASR 1000 Series Router web user interface, and information on configuring and accessing the web user interface.

Related Documentation

This section refers you to other documentation that also might be useful as you configure your Cisco ASR 1000 Series Router. The documentation listed below is available online.

Cisco ASR 1000 Series Routers Documentation

The Cisco ASR 1000 Series has a documentation roadmap that provides listings to a broad range of documentation available for the Cisco ASR 1000 Series Routers. See the *Cisco ASR 1000 Series Router Documentation Roadmap* at:

http://cisco.com/en/US/products/ps9343/products_documentation_roadmaps_list.html

The documentation homepage for the Cisco ASR 1000 Series Routers contains a wide variety of hardware and software information for the Cisco ASR 1000 Series Routers and can be viewed at:

http://cisco.com/en/US/products/ps9343/tsd_products_support_series_home.html

The documentation homepage for Cisco IOS XE contains Cisco IOS XE technology guides and feature documentation and can be viewed at:

http://cisco.com/en/US/products/ps9587/tsd_products_support_series_home.html

For information on commands, see one of the following resources:

- *Cisco IOS XE Software Command References*
- *Command Lookup Tool* (cisco.com login required)

Conventions

This document uses the following conventions:

Convention	Indication
bold font	Commands and keywords and user-entered text appear in **bold** font.

italic font	Document titles, new or emphasized terms, and arguments for which you supply values are in *italic* font.
[ ]	Elements in square brackets are optional.
{x | y | z }	Required alternative keywords are grouped in braces and separated by vertical bars.
[x | y | z]	Optional alternative keywords are grouped in brackets and separated by vertical bars.
string	A nonquoted set of characters. Do not use quotation marks around the string or the string will include the quotation marks.
`courier` font	Terminal sessions and information the system displays appear in `courier` font.
< >	Nonprinting characters such as passwords are in angle brackets.
[ ]	Default responses to system prompts are in square brackets.
!, #	An exclamation point (!) or a pound sign (#) at the beginning of a line of code indicates a comment line.

Note Means *reader take note*.

Tip Means *the following information will help you solve a problem*.

Caution Means *reader be careful*. In this situation, you might perform an action that could result in equipment damage or loss of data.

Timesaver Means *the described action saves time*. You can save time by performing the action described in the paragraph.

Warning **Means reader be warned. In this situation, you might perform an action that could result in bodily injury.**

Obtaining Documentation and Submitting a Service Request

For information on obtaining documentation, submitting a service request, and gathering additional information, see the monthly *What's New in Cisco Product Documentation*, which also lists all new and revised Cisco technical documentation, at:

http://www.cisco.com/en/US/docs/general/whatsnew/whatsnew.html

Subscribe to the *What's New in Cisco Product Documentation* as a Really Simple Syndication (RSS) feed and set content to be delivered directly to your desktop using a reader application. The RSS feeds are a free service and Cisco currently supports RSS version 2.0.

Software Packaging and Architecture

The Cisco ASR 1000 Series Aggregation Services Routers introduce a new software packaging model and architecture.

This chapter discusses this new packaging and architecture and contains the following sections:

- Software Packaging on the Cisco ASR 1000 Series Routers, page 1-1
- Processes Overview, page 1-6
- File Systems on the Cisco ASR 1000 Series Router, page 1-8
- Autogenerated File Directories and Files, page 1-9

Software Packaging on the Cisco ASR 1000 Series Routers

This section covers the following topics:

- ASR 1000 Series Routers Software Overview, page 1-1
- Consolidated Packages, page 1-2
- Individual Software SubPackages Within a Consolidated Package, page 1-3
- Optional Software SubPackages Outside of Consolidated Packages, page 1-4
- Provisioning Files, page 1-4
- ROMmon Image, page 1-5
- File to Upgrade Field Programmable Hardware Devices, page 1-5

ASR 1000 Series Routers Software Overview

The Cisco ASR 1000 Series Routers run using Cisco IOS XE software. Cisco IOS XE is released using consolidated packages and optional subpackages.

Each consolidated package contains a collection of software subpackages. Each software subpackage is an individual software file that controls a different element or elements of the Cisco ASR 1000 Series Router. Each individual software subpackage can be upgraded individually, or all software subpackages for a specific consolidated package can be upgraded as part of a complete consolidated package upgrade. Importantly, IOS (the RPIOS subpackage) is considered one of the seven individual subpackages that makes up a complete consolidated package. For additional information on individual software subpackages, see the "Consolidated Packages" section on page 1-2.

A collection of software subpackages packaged together creates a single consolidated package. A consolidated package allows users to upgrade all individual subpackages on the router with a single software image download. Consolidated packages can be downloaded from Cisco.com; users who want to run the router using individual subpackages that are part of the consolidated package must first download the image from Cisco.com and extract the individual subpackages from the image, which can be done by entering **request platform** command-line interface commands.

Beginning in Cisco IOS XE Release 2.4, support for a supplemental, optional subpackage of type called *sipspawma* is introduced to support the Cisco WebEx Node for ASR 1000 Series shared port adapter (SPA). Optional subpackages are installed similarly to individual subpackages; however, optional subpackages are not bundled as part of a consolidated package like prior support for the individual subpackages, and optional subpackages must be downloaded independently.

See the "Consolidated Packages" section on page 2 and "Individual Software SubPackages Within a Consolidated Package" section on page 1-3 of this book for additional information on extracting individual subpackages from a consolidated package, supplemental optional subpackages, and other information on managing software.

Consolidated Packages

A consolidated package is a single image composed of individual software subpackage files. A single consolidated package file is a bootable file, and the Cisco ASR 1000 Series Router can be run using the consolidated package.

Note Consolidated packages only contain the required individual subpackage files. These packages do not contain supplemental, optional subpackages, such as the "sipspawma" package for the Cisco WebEx Node for ASR 1000 Series.

Each consolidated package also contains a provisioning file. A provisioning file is used for booting in cases where the individual subpackages are extracted from the consolidated package, or optional subpackages are used to run the router. For additional information on the advantages and disadvantages of running a complete consolidated package, see the "Running the Cisco ASR 1000 Series Routers: An Overview" section on page 4-1. For additional information on provisioning files, see the "Provisioning Files" section on page 1-4.

For information about the consolidated packages available in a specific version of Cisco IOS XE, see the release notes for that version of Cisco IOS XE. The Cisco IOS XE Software Release Notes contains the release notes for each version of Cisco IOS XE.

Important Information About Consolidated Packages

The important information about consolidated packages include:

- For each version of a consolidated package, the RPBase, RPControl, ESPBase, SIPSPA, and SIPBase subpackages are identical among consolidated packages.
- For each version of consolidated package, the RPIOS subpackage is always different among consolidated packages.
- A consolidated package file is a bootable file. If the router is configured to run using a the complete consolidated package, boot the router using the consolidated package file. If the router is configured to run using individual subpackages, boot the router using the provisioning file. For additional information on the advantages and disadvantages of running a complete consolidated package, see the "Running the Cisco ASR 1000 Series Routers: An Overview" section on page 4-1. For additional

information on provisioning files, see the "Provisioning Files" section on page 1-4.

- If you need to install optional subpackages, then you must boot the router using the individual subpackage provisioning file method.

Individual Software SubPackages Within a Consolidated Package

provides an overview of the Cisco ASR 1000 Series Routers subpackages and the purpose of each individual subpackage. Every consolidated package will have all of these individual subpackages. To see additional information about each individual subpackages in a particular Cisco IOS XE release, see *Cisco IOS XE Release Notes* for that release.

Table 1-1 Individual SubPackages

SubPackage	Purpose
RPBase	Provides the operating system software for the Route Processor.
RPControl	Controls the control plane processes that interface between the IOS process and the rest of the platform.
RPAccess	Exports processing of restricted components, such as Secure Socket Layer (SSL), Secure Shell (SSH), and other security features.
RPIOS	Provides the Cisco IOS kernel, which is where IOS features are stored and run. Each consolidated package has a different RPIOS.
ESPBase	Provides the ESP operating system and control processes, and the ESP software.
SIPBase	Controls the SIP operating system and control processes.
SIPSPA	Provides the SPA driver and Field Programmable Device (FPD) images.

Important Notes About Individual SubPackages

The important information about individual subpackage include:

- Individual subpackages cannot be downloaded from Cisco.com individually. To get these individual subpackages, users must download a consolidated package and then extract the individual subpackages from the consolidated package using the command-line interface.
- If the router is being run using individual subpackages instead of being run using a complete consolidated package, the router must be booted using a provisioning file. A provisioning file is included in all consolidated packages and is extracted from the image along with the individual subpackages whenever individual subpackages are extracted. For additional information on provisioning files, see the "Provisioning Files" section on page 1-4.

Optional Software SubPackages Outside of Consolidated Packages

Beginning in Cisco IOS XE Release 2.4, the ASR 1000 Series Routers support a new type of subpackage—this is an optional software subpackage that is available as a separate, external package that is downloaded and installed along with the other required subpackages.

sipspawmak9 is an optional subpackage that provides the system software for the Cisco WebEx Node for ASR 1000 Series Routers.

Important Notes About Optional SubPackages

The important information about optional subpackages include:

- Optional subpackages are downloaded separately from consolidated package files. Optional subpackages are not contained within a consolidated package for a release.
- Optional package installation works similarly to the installation of individual subpackages using a provisioning file.
- Optional subpackages can be uninstalled to remove provisioning when the package no longer applies to an RP.
- Optional subpackages are easily supported by the standard ISSU upgrade process as long as the package is located in the directory of the provisioning file for each RP.

Provisioning Files

Note You must use the provisioning files to manage the boot process if you need to install optional subpackages.

Provisioning files manage the boot process when the Cisco ASR 1000 Series Router is configured to run using individual subpackages or optional subpackages (such as the package for the Cisco WebEx Node for ASR 1000 Series). When individual subpackages are being used to run the Cisco ASR 1000 Series Router, the router has to be configured to boot the provisioning file. The provisioning file manages the bootup of each individual subpackage and the Cisco ASR 1000 Series Router assumes normal operation.

Provisioning files are extracted automatically when individual subpackage files are extracted from a consolidated package.

Provisioning files are not necessary for running the router using the complete consolidated package; if you want to run the router using the complete consolidated package, simply boot the router using the consolidated package file.

See the "Running the Cisco ASR 1000 Series Routers: An Overview" section on page 4-1 for additional information on the advantages and disadvantages of running individual subpackages versus running a complete consolidated package.

Important Notes About Provisioning Files

The important information about provisioning files include:

- Each consolidated package contains two provisioning files. One of the provisioning files is always named “packages.conf”, while the other provisioning file will have a name based on the consolidated package naming structure. In any consolidated package, both provisioning files perform the exact same function.
- In most cases, the “packages.conf” provisioning file should be used to boot the router. Configuring the router to boot using this file is generally easier because the router can be configured to boot using “packages.conf”, so no changes have to be made to the boot statement when Cisco IOS XE is upgraded (the **boot system** *file-system*:**packages.conf** configuration command can remain unmodified before and after an upgrade).
- The provisioning file and individual subpackage files must be kept in the same directory. The provisioning file does not work properly if the individual subpackage files are in other directories.
- The provisioning filename can be renamed; the individual subpackage filenames cannot be renamed.
- After placing the provisioning file and the individual subpackage files in a directory and booting the router, it is highly advisable not to rename, delete, or alter any of these files. Renaming, deleting, or altering the files can lead to unpredictable router problems and behaviors.

ROMmon Image

An independent ROMmon image is released periodically separate from consolidated packages or any other software releases.

See the documentation that accompanies the ROMmon image for information on each ROMmon image. For additional information on ROMmon, see the *Cisco ASR 1000 Series Routers Maintain and Operate Guide*.

File to Upgrade Field Programmable Hardware Devices

Starting in Cisco IOS XE Release 3.1.0S, a hardware programmable package file used to upgrade field programmable hardware devices is released as needed. A package file is provided for the field programmable device to customers in cases where a field upgrade is required. If the Cisco ASR 1000 Series Router contains an incompatible version of the hardware programmable firmware on the Cisco ASR1000-RP, Cisco ASR1000-SIP, or Cisco ASR1000-ESP, then that firmware may need to be upgraded.

Generally an upgrade is only necessary in cases where a system message indicates one of the field programmable devices on the Cisco ASR 1000 Series Router needs an upgrade or a Cisco technical support representative suggests an upgrade.

In Cisco IOS XE Release 3.1.0S, a package file that contains a new version of the Complex Programmable Logic Device (CPLD) code is available for users who need to upgrade old versions of firmware on a Cisco ASR1000-RP2 or Cisco ASR1000-SIP10 in a Cisco ASR 1013 Router.

For more information on upgrading field programmable hardware devices, see the *Upgrading Field Programmable Hardware Devices for Cisco ASR 1000 Series Routers* document.

Processes Overview

Cisco IOS XE has numerous components that run entirely as separate processes on the Cisco ASR 1000 Series Routers. This modular architecture increases network resiliency by distributing operating responsibility among separate processes rather than relying on Cisco IOS software for all operations.

This section covers the following topics:

- IOS as a Process, page 1-7
- Dual IOS Processes, page 1-8

Table 1-2 provides a list of the important individual processes for the Cisco ASR 1000 Series Routers. These processes run in the background, and the CLI on Cisco ASR 1000 Series Routers using Cisco IOS-XE is identical in look, feel, and usage to the Cisco IOS CLI on most platforms. This information is provided because it may be useful for checking router state and troubleshooting, but understanding this information is not essential to understanding most router operations.

Table 1-2 Individual Processes

Process	Purpose	Affected FRUs	SubPackage Mapping
Chassis Manager	Responsible for all chassis management functions, including management of the HA state, environmental monitoring, and FRU state control.	RP (one instance per RP) SIP (one instance per SIP) ESP (one instance per ESP)	RPControl SIPBase ESPBase
Host Manager	Provides an interface between the IOS process and many of the information-gathering functions of the underlying platform kernel and operating system.	RP (one instance per RP) SIP (one instance per SIP) ESP (one instance per ESP)	RPControl SIPBase ESPBase
Logger	Provides IOS facing logging services to processes running on each FRU.	RP (one instance per RP) SIP (one instance per SIP) ESP (one instance per ESP)	RPControl SIPBase ESPBase
Interface Manager	Provides an interface between the IOS process and the per-SPA interface processes on the SIP.	RP (one instance per RP) SIP (one instance per SIP)	RPControl SIPBase
IOS	The IOS process implements all forwarding and routing features for the router.	RP (one per software redundancy instance per RP). Maximum of two instances per RP.	RPIOS

Table 1-2 Individual Processes (continued)

Process	Purpose	Affected FRUs	SubPackage Mapping
Forwarding Manager	Manages the downloading of configuration to each of the ESPs and the communication of forwarding plane information, such as statistics, to the IOS process.	RP (one per software redundancy instance per RP). Maximum of two instances per RP. ESP (one per ESP)	RPControl ESPBase
Pluggable Services	The integration point between platform policy application, such as authentication and the IOS process.	RP (one per software redundancy instance per RP). Maximum of two instances per RP.	RPControl
Shell Manager	Provides all user interface features and handling related to features in the nonIOS image of the consolidated package, which are also the features available in diagnostic mode when the IOS process fails.	RP (one instance per RP)	RPControl
SPA driver process	Provides an isolated process driver for a specific SPA.	SPA (one instance per SPA per SIP)	SIPSPA
CPP driver process	Manages the CPP hardware forwarding engine on the ESP.	ESP (one instance per ESP)	ESPBase
CPP HA process	Manages HA state for the CPP hardware forwarding engine.	ESP (one instance per ESP)	ESPBase
CPP SP process	Performs high-latency tasks for the CPP-facing functionality in the ESP instance of the Forwarding Manager process.	ESP (one instance per ESP)	ESPBase

IOS as a Process

In almost all previous Cisco router platforms, an overwhelming majority of the internal software processes are run using Cisco IOS memory.

The Cisco ASR 1000 Series Routers introduce a distributed software architecture that moves many operating system responsibilities out of the IOS process. In this architecture, IOS, which previously was responsible for almost all of the internal software processes, now runs as one of many Linux processes while allowing other Linux processes to share responsibility for running the router.

This architecture allows for better allocation of memory so the router can run more efficiently. Furthermore, the Cisco ASR 1000 Series Routers has a command-line interface in diagnostic mode that can be accessed in certain scenarios even if the IOS process fails. See the "Understanding the Diagnostic Mode" section on page 2-7 for information on diagnostic mode.

Dual IOS Processes

The Cisco ASR 1000 Series Router introduces a dual IOS process model that allows for increased high availability at all times.

Using SSO or RPR, a second IOS process can be enabled on a Cisco ASR 1002 or 1004 Router. On Cisco ASR 1000 Series Routers configured with dual Route Processors, the second IOS process runs on the standby Route Processor.

The state of these dual IOS processes can be checked by entering the **show platform** command.

The advantages of a second IOS process includes:

- Increased fault tolerance—In the event of an active IOS failure, the second IOS process immediately becomes the active IOS process with little to no service disruption.
- No downtime software upgrades—IOS and other software on the router can be upgraded using the In Service Software Upgrade (ISSU) feature in the standby IOS process, thereby allowing the network to remain active during the software upgrade. See the "Router#" section on page 4-20 for additional information on when ISSU can and cannot be used to perform no downtime software upgrades.

File Systems on the Cisco ASR 1000 Series Router

Table 1-3 provides a list of file systems that can be seen on the Cisco ASR 1000 Series Routers.

Table 1-3 File Systems

File System	Description
bootflash:	The boot flash memory file system on the active RP.
cns:	The Cisco Networking Services file directory.
harddisk:	The hard disk file system on the active RP. The harddisk: file system is not available on the Cisco ASR 1002 Routers.
nvram:	Router NVRAM. You can copy the startup configuration to NVRAM or from NVRAM.
obfl:	The file system for Onboard Failure Logging files.
stby-bootflash:	The boot flash memory file system on the standby RP.
stby-harddisk:	The hard disk file system on the standby RP. The harddisk: file system is not available on the Cisco ASR 1002 Routers.
stby-usb[0-1]:	The Universal Serial Bus (USB) flash drive file systems on the standby RP. The stby-usb: file system is not available on the Cisco ASR 1002 Routers.
system:	The system memory file system, which includes the running configuration.
tar:	The archive file system.
tmpsys:	The temporary system files file system.
usb[0-1]:	The Universal Serial Bus (USB) flash drive file systems on the active RP. Only usb0: is available on the Cisco ASR 1002 Router.

If you run into a file system not listed in Table 1-3, enter the **?** help option or see the **copy** command reference for additional information on that file system.

Autogenerated File Directories and Files

This section discusses the autogenerated files and directories that might appear on your Cisco ASR 1000 Series Routers, and how the files in these directories can be managed.

Table 1-4 provides a list and descriptions of autogenerated files on the Cisco ASR 1000 Series Routers.

Table 1-4 Autogenerated Files

File or Directory	Description
crashinfo files	A crashinfo file may appear in the bootflash: or harddisk: file system. These files provide descriptive information of a crash and may be useful for tuning or troubleshooting purposes, but the files are not part of router operations and can be erased without impacting the functioning of the router.
core directory	The storage area for.core files. If this directory is erased, it will automatically regenerate itself at bootup. The .core files in this directory can be erased without impacting any router functionality, but the directory itself should not be erased.
lost+found directory	This directory is created on bootup if a system check is performed. Its appearance is completely normal and does not indicate any issues with the router.
tracelogs directory	The storage area for trace files. Trace files are useful for troubleshooting; if the IOS process fails, for instance, users or troubleshooting personnel can access trace files using diagnostic mode to gather information related to the IOS failure. Trace files, however, are not part of router operations and can be erased without impacting the router's performance.

Important Notes About Autogenerated Directories

The important information about autogenerated directories include:

- Any autogenerated file on the bootflash: directory should not be deleted, renamed, moved, or altered in any way unless directed by customer support. Altering autogenerating files on the bootflash: can have unpredictable consequences for system performance.
- Crashinfo, core, and trace files can be deleted, but the core and tracelog directories that are automatically part of the harddisk: file system should not be deleted.

Using Cisco IOS XE Software

This chapter provides information to prepare you to configure the Cisco ASR 100Series Routers:

- Accessing the CLI Using a Router Console, page 2-1
- Using the Auxiliary Port, page 2-4
- Using Keyboard Shortcuts, page 2-5
- Using the History Buffer to Recall Commands, page 2-5
- Understanding the Command Mode, page 2-6
- Understanding the Diagnostic Mode, page 2-7
- Getting Help, page 2-8
- Using the no and default Forms of Commands, page 2-11
- Saving Configuration Changes, page 2-11
- Managing Configuration Files, page 2-11
- Filtering the Output of the show and more Commands, page 2-13
- Powering Off a Router, page 2-14
- Finding Support Information for Platforms and Cisco Software Images, page 2-14

Accessing the CLI Using a Router Console

The following sections describe how to access the command-line interface (CLI) using a directly-connected console or by using Telnet or a modem to obtain a remote console:

- Accessing the CLI Using a Directly-Connected Console, page 2-2
- Accessing the CLI from a Remote Console Using Telnet, page 2-3
- Accessing the CLI from a Remote Console Using a Modem, page 2-4

Accessing the CLI Using a Directly-Connected Console

This section describes how to connect to the console port on the router and use the console interface to access the CLI.

The console port on a Cisco ASR 1000 Series Router is an EIA/TIA-232 asynchronous, serial connection with no flow control and an RJ-45 connector. The console port is located on the front panel of each Route Processor (RP).

Connecting to the Console Port

To connect to the console port, complete the following steps:

Step 1 Configure your terminal emulation software with the following settings:

- 9600 bits per second (bps)
- 8 data bits
- No parity
- 1 stop bit
- No flow control

Step 2 Connect to the port using the RJ-45-to-RJ-45 cable and RJ-45-to-DB-25 DTE adapter or using the RJ-45-to-DB-9 DTE adapter (labeled “Terminal”).

Using the Console Interface

Every RP on a Cisco ASR 1000 Series Router has a console interface. Notably, a standby RP can be accessed using the console port in addition to the active RP in a dual RP configuration.

To access the CLI using the console interface, complete the following steps:

Step 1 After you attach the terminal hardware to the console port on the router and you configure your terminal emulation software with the proper settings, the following prompt appears:

```
Press RETURN to get started.
```

Step 2 Press **Return** to enter user EXEC mode. The following prompt appears:

```
Router>
```

Step 3 From user EXEC mode, enter the **enable** command as shown in the following example:

```
Router> enable
```

Step 4 At the password prompt, enter your system password. If an enable password has not been set on your system, this step may be skipped.
The following example shows entry of the password enablepass:

```
Password: enablepass
```

Step 5 When your enable password is accepted, the privileged EXEC mode prompt appears:

```
Router#
```

Step 6 You now have access to the CLI in privileged EXEC mode and you can enter the necessary commands to complete your desired tasks.

Step 7 To exit the console session, enter the **quit** command as shown in the following example:

```
Router# quit
```

Accessing the CLI from a Remote Console Using Telnet

This section describes how to connect to the console interface on a router using Telnet to access the CLI.

Preparing to Connect to the Router Console Using Telnet

Before you can access the router remotely using Telnet from a TCP/IP network, you need to configure the router to support virtual terminal lines (vtys) using the **line vty** global configuration command. You also should configure the vtys to require login and specify a password.

Note To prevent disabling login on the line, be careful that you specify a password with the **password** command when you configure the **login** line configuration command. If you are using authentication, authorization, and accounting (AAA), you should configure the **login authentication** line configuration command. To prevent disabling login on the line for AAA authentication when you configure a list with the **login authentication** command, you must also configure that list using the **aaa authentication login** global configuration command. For more information about AAA services, see the *Cisco IOS XE Security Configuration Guide,* and the *Cisco IOS Security Command Reference Guide*.

In addition, before you can make a Telnet connection to the router, you must have a valid host name for the router or have an IP address configured on the router. For more information about requirements for connecting to the router using Telnet, information about customizing your Telnet services, and using Telnet key sequences, see the *Cisco IOS Configuration Fundamentals Configuration Guide.*

Using Telnet to Access a Console Interface

To access a console interface using Telnet, complete the following steps:

Step 1 From your terminal or PC, enter one of the following commands:

- **connect** *host* [*port*] [*keyword*]
- **telnet** *host* [*port*] [*keyword*]

In this syntax, *host* is the router hostname or an IP address, *port* is a decimal port number (23 is the default), and *keyword* is a supported keyword. For more information, see the *Cisco IOS Configuration Fundamentals Command Reference Guide*.

Note If you are using an access server, then you will need to specify a valid port number such as **telnet 172.20.52.40 2004**, in addition to the hostname or IP address.

The following example shows the **telnet** command to connect to the router named router:

```
unix_host% telnet router
```

```
Trying 172.20.52.40...
Connected to 172.20.52.40.
Escape character is '^]'.
unix_host% connect
```

Step 2 At the password prompt, enter your login password. The following example shows entry of the password mypass:

```
User Access Verification

Password: mypass
```

Note If no password has been configured, press **Return**.

Step 3 From user EXEC mode, enter the **enable** command as shown in the following example:

```
Router> enable
```

Step 4 At the password prompt, enter your system password. The following example shows entry of the password enablepass:

```
Password: enablepass
```

Step 5 When the enable password is accepted, the privileged EXEC mode prompt appears:

```
Router#
```

Step 6 You now have access to the CLI in privileged EXEC mode and you can enter the necessary commands to complete your desired tasks.

Step 7 To exit the Telnet session, use the **exit** or **logout** command as shown in the following example:

```
Router# logout
```

Accessing the CLI from a Remote Console Using a Modem

To access the router remotely using a modem through an asynchronous connection, connect the modem to the console port.

The console port on a Cisco ASR 1000 Series Aggregation Services Router is an EIA/TIA-232 asynchronous, serial connection with no flow control and an RJ-45 connector. The console port is located on the front panel of the RP.

To connect a modem to the console port, place the console port mode switch in the in position. Connect to the port using the RJ-45-to-RJ-45 cable and the RJ-45-to-DB-25 DCE adapter (labeled “Modem”).

Using the Auxiliary Port

The auxiliary port on the Route Processor does not serve any useful purpose for customers.

This port should only be accessed under the advisement of a customer support representative.

Using Keyboard Shortcuts

Commands are not case sensitive. You can abbreviate commands and parameters if the abbreviations contain enough letters to be different from any other currently available commands or parameters.

Table 2-1 lists the keyboard shortcuts for entering and editing commands.

Table 2-1 Keyboard Shortcuts

Keystrokes	Purpose
Ctrl-B or the **Left Arrow** key[1]	Move the cursor back one character
Ctrl-F or the **Right Arrow** key1	Move the cursor forward one character
Ctrl-A	Move the cursor to the beginning of the command line
Ctrl-E	Move the cursor to the end of the command line
Esc B	Move the cursor back one word
Esc F	Move the cursor forward one word

1. The arrow keys function only on ANSI-compatible terminals such as VT100s.

Using the History Buffer to Recall Commands

The history buffer stores the last 20 commands you entered. History substitution allows you to access these commands without retyping them, by using special abbreviated commands.

Table 2-2 lists the history substitution commands.

Table 2-2 History Substitution Commands

Command	Purpose
Ctrl-P or the **Up Arrow** key[1]	Recall commands in the history buffer, beginning with the most recent command. Repeat the key sequence to recall successively older commands.
Ctrl-N or the **Down Arrow** key1	Return to more recent commands in the history buffer after recalling commands with **Ctrl-P** or the **Up Arrow** key.
Router# **show history**	While in EXEC mode, list the last several commands you have just entered.

1. The arrow keys function only on ANSI-compatible terminals such as VT100s.

Understanding the Command Mode

The command modes available in the traditional Cisco IOS CLI are exactly the same as the command modes available in Cisco IOS XE.

You use the CLI to access Cisco IOS XE software. Because the CLI is divided into many different modes, the commands available to you at any given time depend on the mode that you are currently in. Entering a question mark (**?**) at the CLI prompt allows you to obtain a list of commands available for each command mode.

When you log in to the CLI, you are in user EXEC mode. User EXEC mode contains only a limited subset of commands. To have access to all commands, you must enter privileged EXEC mode, normally by using a password. From privileged EXEC mode, you can issue any EXEC command—user or privileged mode—or you can enter global configuration mode. Most EXEC commands are one-time commands. For example, **show** commands show important status information, and **clear** commands clear counters or interfaces. The EXEC commands are not saved when the software reboots.

Configuration modes allow you to make changes to the running configuration. If you later save the running configuration to the startup configuration, these changed commands are stored when the software is rebooted. To enter specific configuration modes, you must start at global configuration mode. From global configuration mode, you can enter interface configuration mode and a variety of other modes, such as protocol-specific modes.

ROM monitor mode is a separate mode used when the Cisco IOS XE software cannot load properly. If a valid software image is not found when the software boots or if the configuration file is corrupted at startup, the software might enter ROM monitor mode.

Table 2-3 describes how to access and exit various common command modes of the Cisco IOS XE software. It also shows examples of the prompts displayed for each mode.

Table 2-3 Accessing and Exiting Command Modes

Command Mode	Access Method	Prompt	Exit Method
User EXEC	Log in.	`Router>`	Use the **logout** command.
Privileged EXEC	From user EXEC mode, use the **enable** EXEC command.	`Router#`	To return to user EXEC mode, use the **disable** command.
Global configuration	From privileged EXEC mode, use the **configure terminal** privileged EXEC command.	`Router(config)#`	To return to privileged EXEC mode from global configuration mode, use the **exit** or **end** command.
Interface configuration	From global configuration mode, specify an interface using an **interface** command.	`Router(config-if)#`	To return to global configuration mode, use the **exit** command. To return to privileged EXEC mode, use the **end** command.

Table 2-3 Accessing and Exiting Command Modes (continued)

Command Mode	Access Method	Prompt	Exit Method
Diagnostic	The router boots up or accesses diagnostic mode in the following scenarios: In some cases, diagnostic mode will be reached when the IOS process or processes fail. In most scenarios, however, the router will. A user-configured access policy was configured using the **transport-map** command that directed the user into diagnostic mode. See the “”chapter of this book for information on configuring access policies. The router was accessed using a Route Processor auxiliary port. A break signal (**Ctrl-C**, **Ctrl-Shift-6**, or the **send break** command) was entered and the router was configured to go into diagnostic mode when the break signal was received.	`Router(diag)#`	If the IOS process failing is the reason for entering diagnostic mode, the IOS problem must be resolved and the router rebooted to get out of diagnostic mode. If the router is in diagnostic mode because of a transport-map configuration, access the router through another port or using a method that is configured to connect to the Cisco IOS CLI. If the router is accessed through the Route Processor auxiliary port, access the router through another port. Accessing the router through the auxiliary port is not useful for customer purposes anyway.
ROM monitor	From privileged EXEC mode, use the **reload** EXEC command. Press the **Break** key during the first 60 seconds while the system is booting.	`>`	To exit ROM monitor mode, use the **continue** command.

Understanding the Diagnostic Mode

Diagnostic mode is new on the Cisco ASR 1000 Series Routers.

The router boots up or accesses diagnostic mode in the following scenarios:

- The IOS process or processes fail, in some scenarios. In other scenarios, the RP will simply reset when the IOS process or processes fail.
- A user-configured access policy was configured using the **transport-map** command that directs the user into diagnostic mode.
- The router was accessed using a Route Processor auxiliary port.
- A send break signal (**Ctrl-C** or **Ctrl-Shift-6**) was entered while accessing the router, and the router was configured to enter diagnostic mode when a break signal was sent.

In diagnostic mode, a subset of the commands that are also available in User EXEC mode are made available to users. Among other things, these commands can be used to:

- Inspect various states on the router, including the IOS state.
- Replace or roll back the configuration.
- Provide methods of restarting the IOS or other processes.
- Reboot hardware, such as the entire router, an RP, an ESP, a SIP, a SPA, or possibly other hardware components.
- Transfer files into or off of the router using remote access methods such as FTP, TFTP, SCP, and so on.

The diagnostic mode provides a more comprehensive user interface for troubleshooting than previous routers, which relied on limited access methods during failures, such as ROMmon, to diagnose and troubleshoot IOS problems.

The diagnostic mode commands are stored in the nonIOS packages on the Cisco ASR 1000 Series Routers, which is why the commands are available even if the IOS process is not working properly. Importantly, all the commands available in diagnostic mode are also available in privileged EXEC mode on the router even during normal router operation. The commands are entered like any other commands in the privileged EXEC command prompts when used in privileged EXEC mode.

Getting Help

Entering a question mark (**?**) at the CLI prompt displays a list of commands available for each command mode. You can also get a list of keywords and arguments associated with any command by using the context-sensitive help feature.

To get help specific to a command mode, a command, a keyword, or an argument, use one of the commands listed in Table 2-4:

Table 2-4 Help Commands and Purpose

Command	Purpose
help	Provides a brief description of the help system in any command mode.
abbreviated-command-entry?	Provides a list of commands that begin with a particular character string. (No space between command and question mark.)
abbreviated-command-entry<Tab>	Completes a partial command name.
?	Lists all commands available for a particular command mode.
command ?	Lists the keywords or arguments that you must enter next on the command line. (Space between command and question mark.)

Finding Command Options

This section provides an example of how to display syntax for a command. The syntax can consist of optional or required keywords and arguments. To display keywords and arguments for a command, enter a question mark (**?**) at the configuration prompt or after entering part of a command followed by a space. The Cisco IOS XE software displays a list and brief description of available keywords and arguments. For example, if you were in global configuration mode and wanted to see all the keywords or arguments for the **arap** command, you would type **arap ?**.

The <cr> symbol in command help output stands for “carriage return.” On older keyboards, the carriage return key is the Return key. On most modern keyboards, the carriage return key is the Enter key. The <cr> symbol at the end of command help output indicates that you have the option to press **Enter** to complete the command and that the arguments and keywords in the list preceding the <cr> symbol are optional. The <cr> symbol by itself indicates that no more arguments or keywords are available and that you must press **Enter** to complete the command.

Table 2-5 shows examples of how you can use the question mark (**?**) to assist you in entering commands.

Table 2-5 Finding Command Options

Command	Comment

`Router> enable`
`Password: <password>`
`Router#`
	Enter the **enable** command and password to access privileged EXEC commands. You are in privileged EXEC mode when the prompt changes to a "#" from the ">"; for example, `Router> to Router#`.

`Router# configure terminal`
`Enter configuration commands, one per line. End with CNTL/Z.`
`Router(config)#`
	Enter the **configure terminal** privileged EXEC command to enter global configuration mode. You are in global configuration mode when the prompt changes to `Router(config)#`.

`Router(config)# interface serial ?`
`  <0-6>  Serial interface number`
`Router(config)# interface serial 4 ?`
`  /`
`Router(config)# interface serial 4/ ?`
`  <0-3>  Serial interface number`
`Router(config)# interface serial 4/0 ?`
`<cr>`
`Router(config)# interface serial 4/0`
`Router(config-if)#`
	Enter interface configuration mode by specifying the serial interface that you want to configure using the **interface serial** global configuration command.
	Enter **?** to display what you must enter next on the command line. In this example, you must enter the serial interface slot number and port number, separated by a forward slash.
	When the <cr> symbol is displayed, you can press **Enter** to complete the command.
	You are in interface configuration mode when the prompt changes to `Router(config-if)#`.

`Router(config-if)# ?`
`Interface configuration commands:`
`  .`
`  .`
`  .`
`  ip  Interface Internet Protocol config commands`
`  keepalive  Enable keepalive`
`  lan-name  LAN Name command`
`  llc2  LLC2 Interface Subcommands`
`  load-interval  Specify interval for load calculation for an interface`
`  locaddr-priority  Assign a priority group`
`  logging  Configure logging for interface`
`  loopback  Configure internal loopback on an interface`
`  mac-address  Manually set interface MAC address`
`  mls  mls router sub/interface commands`
`  mpoa  MPOA interface configuration commands`
`  mtu  Set the interface Maximum Transmission Unit (MTU)`
`  netbios  Use a defined NETBIOS access list or enable name-caching`
`  no  Negate a command or set its defaults`
`  nrzi-encoding  Enable use of NRZI encoding`
`  ntp  Configure NTP`
`  .`
`  .`
`  .`
`Router(config-if)#`
	Enter **?** to display a list of all the interface configuration commands available for the serial interface. This example shows only some of the available interface configuration commands.

Table 2-5 Finding Command Options (continued)

Command	Comment

`Router(config-if)# ip ?`
`Interface IP configuration subcommands:`
`  access-group  Specify access control for packets`
`  accounting  Enable IP accounting on this interface`
`  address  Set the IP address of an interface`
`  authentication  authentication subcommands`
`  bandwidth-percent  Set EIGRP bandwidth limit`
`  broadcast-address  Set the broadcast address of an interface`
`  cgmp  Enable/disable CGMP`
`  directed-broadcast  Enable forwarding of directed broadcasts`
`  dvmrp  DVMRP interface commands`
`  hello-interval  Configures IP-EIGRP hello interval`
`  helper-address  Specify a destination address for UDP broadcasts`
`  hold-time  Configures IP-EIGRP hold time`
`  .`
`  .`
`  .`
`Router(config-if)# ip`
	Enter the command that you want to configure for the interface. This example uses the **ip** command.
	Enter **?** to display what you must enter next on the command line. This example shows only some of the available interface IP configuration commands.

`Router(config-if)# ip address ?`
`  A.B.C.D  IP address`
`  negotiated  IP Address negotiated over PPP`
`Router(config-if)# ip address`
	Enter the command that you want to configure for the interface. This example uses the **ip address** command.
	Enter **?** to display what you must enter next on the command line. In this example, you must enter an IP address or the **negotiated** keyword.
	A carriage return (<cr>) is not displayed; therefore, you must enter additional keywords or arguments to complete the command.

`Router(config-if)# ip address 172.16.0.1 ?`
`  A.B.C.D  IP subnet mask`
`Router(config-if)# ip address 172.16.0.1`
	Enter the keyword or argument that you want to use. This example uses the 172.16.0.1 IP address.
	Enter **?** to display what you must enter next on the command line. In this example, you must enter an IP subnet mask.
	A <cr> is not displayed; therefore, you must enter additional keywords or arguments to complete the command.

`Router(config-if)# ip address 172.16.0.1 255.255.255.0 ?`
`  secondary  Make this IP address a secondary address`
`  <cr>`
`Router(config-if)# ip address 172.16.0.1 255.255.255.0`
	Enter the IP subnet mask. This example uses the 255.255.255.0 IP subnet mask.
	Enter **?** to display what you must enter next on the command line. In this example, you can enter the **secondary** keyword, or you can press **Enter**.
	A <cr> is displayed; you can press **Enter** to complete the command, or you can enter another keyword.

`Router(config-if)# ip address 172.16.0.1 255.255.255.0` `Router(config-if)#`	In this example, **Enter** is pressed to complete the command.

Using the no and default Forms of Commands

Almost every configuration command has a **no** form. In general, use the **no** form to disable a function. Use the command without the **no** keyword to re-enable a disabled function or to enable a function that is disabled by default. For example, IP routing is enabled by default. To disable IP routing, use the **no ip routing** command; to re-enable IP routing, use the **ip routing** command. The Cisco IOS software command reference publications provide the complete syntax for the configuration commands and describe what the **no** form of a command does.

Many CLI commands also have a **default** form. By issuing the command **default** *command-name*, you can configure the command to its default setting. The Cisco IOS software command reference publications describe the function of the **default** form of the command when the **default** form performs a different function than the plain and **no** forms of the command. To see what default commands are available on your system, enter **default ?** in the appropriate command mode.

Saving Configuration Changes

Use the **copy running-config startup-config** command to save your configuration changes to the startup configuration so that the changes will not be lost if the software reloads or a power outage occurs. For example:

```
Router# copy running-config startup-config
Building configuration...
```

It might take a minute or two to save the configuration. After the configuration has been saved, the following output appears:

```
[OK]
Router#
```

This task saves the configuration to NVRAM.

Managing Configuration Files

On the Cisco ASR 1000 Series Routers, the startup configuration file is stored in the nvram: file system and the running-configuration files are stored in the system: file system. This configuration file storage setup is not unique to the Cisco ASR 1000 Series Routers and is used on several Cisco router platforms.

As a matter of routine maintenance on any Cisco router, users should backup the startup configuration file by copying the startup configuration file from NVRAM onto one of the router's other file systems and, additionally, onto a network server. Backing up the startup configuration file provides an easy method of recovering the startup configuration file in the event the startup configuration file in NVRAM becomes unusable for any reason.

The **copy** command can be used to backup startup configuration files. The following examples show the startup configuration file in NVRAM being backed up:

Example 1: Copying a Startup Configuration File to Bootflash

```
Router# dir bootflash:
Directory of bootflash:/

   11  drwx       16384   Dec 4 2007 04:32:46 -08:00  lost+found
86401  drwx        4096   Dec 4 2007 06:06:24 -08:00  .ssh
14401  drwx        4096   Dec 4 2007 06:06:36 -08:00  .rollback_timer
```

```
28801  drwx        4096  May 29 2008 16:31:41 -07:00  .prst_sync
43201  drwx        4096   Dec 4 2007 04:34:45 -08:00  .installer
   12  -rw-   208904396  May 28 2008 16:17:34 -07:00
asr1000rp1-adventerprisek9.02.01.00.122-33.XNA.bin

Router# copy nvram:startup-config bootflash:
Destination filename [startup-config]?

3517 bytes copied in 0.647 secs (5436 bytes/sec)

Router# dir bootflash:
Directory of bootflash:/

   11  drwx       16384   Dec 4 2007 04:32:46 -08:00  lost+found
86401  drwx        4096   Dec 4 2007 06:06:24 -08:00  .ssh
14401  drwx        4096   Dec 4 2007 06:06:36 -08:00  .rollback_timer
28801  drwx        4096  May 29 2008 16:31:41 -07:00  .prst_sync
43201  drwx        4096   Dec 4 2007 04:34:45 -08:00  .installer
   12  -rw-   208904396  May 28 2008 16:17:34 -07:00
asr1000rp1-adventerprisek9.02.01.00.122-33.XNA.bin
13 -rw-        7516   Jul 2 2008 15:01:39 -07:00  startup-config
```

Example 2: Copying a Startup Configuration File to USB Flash Disk

```
Router# dir usb0:
Directory of usb0:/

43261  -rwx   208904396  May 27 2008 14:10:20 -07:00
asr1000rp1-adventerprisek9.02.01.00.122-33.XNA.bin

255497216 bytes total (40190464 bytes free)

Router# copy nvram:startup-config usb0:
Destination filename [startup-config]?

3172 bytes copied in 0.214 secs (14822 bytes/sec)

Router# dir usb0:
Directory of usb0:/

43261  -rwx   208904396  May 27 2008 14:10:20 -07:00
asr1000rp1-adventerprisek9.02.01.00.122-33.XNA.bin
43262 -rwx        3172   Jul 2 2008 15:40:45 -07:00  startup-config

255497216 bytes total (40186880 bytes free)
```

Example 3: Copying a Startup Configuration File to a TFTP Server

```
Router# copy bootflash:startup-config tftp:
Address or name of remote host []? 172.17.16.81
Destination filename [pe24_asr-1002-confg]? /auto/tftp-users/user/startup-config
!!
3517 bytes copied in 0.122 secs (28828 bytes/sec)
```

For more detailed information on managing configuration files, see the *Managing Configuration Files* section in the *Cisco IOS XE Configuration Fundamentals Configuration Guide*.

Filtering the Output of the show and more Commands

You can search and filter the output of **show** and **more** commands. This functionality is useful if you need to sort through large amounts of output or if you want to exclude output that you need not see.

To use this functionality, enter a **show** or **more** command followed by the "pipe" character (|); one of the keywords **begin**, **include**, or **exclude**; and a regular expression on which you want to search or filter (the expression is case sensitive):

show *command* | {**append** | **begin** | **exclude** | **include** | **redirect** | **section** | **tee**} *regular-expression*

The output matches certain lines of information in the configuration file. The following example illustrates how to use output modifiers with the **show interface** command when you want the output to include only lines in which the expression "protocol" appears:

```
Router# show interface | include protocol

FastEthernet0/0 is up, line protocol is up
Serial4/0 is up, line protocol is up
Serial4/1 is up, line protocol is up
Serial4/2 is administratively down, line protocol is down
Serial4/3 is administratively down, line protocol is down
```

Powering Off a Router

Before you turn off a power supply, make certain the chassis is grounded and you perform a soft shutdown on the power supply. Not performing a soft shutdown will often not harm the router, but may cause problems in certain scenarios.

To perform a soft shutdown before powering off the router, enter the **reload** command to halt the system and then wait for ROM Monitor to execute before proceeding to the next step.

The following screenshot shows an example of this process:

```
Router# reload
Proceed with reload? [confirm]
...(Some messages are omitted here)
Initializing Hardware...

Calculating the ROMMON CRC...CRC is correct.
```

Place the power supply switch in the Off position after seeing this message.

Finding Support Information for Platforms and Cisco Software Images

Cisco software is packaged in feature sets consisting of software images that support specific platforms. The feature sets available for a specific platform depend on which Cisco software images are included in a release. To identify the set of software images available in a specific release or to find out if a feature is available in a given Cisco IOS XE software image, you can use Cisco Feature Navigator or the software release notes.

Using the Cisco Feature Navigator

Use Cisco Feature Navigator to find information about platform support and software image support. Cisco Feature Navigator enables you to determine which Cisco IOS XE software images support a specific software release, feature set, or platform. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Using the Software Advisor

To see if a feature is supported by a Cisco IOS XE release, to locate the software document for that feature, or to check the minimum software requirements of Cisco IOS XE software with the hardware installed on your router, Cisco maintains the Software Advisor tool on Cisco.com at http://www.cisco.com/cgi-bin/Support/CompNav/Index.pl.

You must be a registered user on Cisco.com to access this tool.

Using the Software Release Notes

Cisco IOS XE software releases include release notes that provide the following information:

- Platform support information
- Memory recommendations
- New feature information
- Open and resolved severity 1 and 2 caveats for all platforms

Release notes are intended to be release-specific for the most current release, and the information provided in these documents may not be cumulative in providing information about features that first appeared in previous releases. Refer to Cisco Feature Navigator for cumulative feature information.

Console Port, Telnet, and SSH Handling

This chapter covers the following topics:

- Console Port Overview for the Cisco ASR 1000 Series Routers, page 3-1
- Console Port Handling Overview, page 3-1
- Telnet and SSH Overview for the Cisco ASR 1000 Series Routers, page 3-2
- Persistent Telnet and Persistent SSH Overview, page 3-2
- Configuring a Console Port Transport Map, page 3-3
- Configuring Persistent Telnet, page 3-5
- Configuring Persistent SSH, page 3-8
- Viewing Console Port, SSH, and Telnet Handling Configurations, page 3-11
- Important Notes and Restrictions, page 3-16

Console Port Overview for the Cisco ASR 1000 Series Routers

The console port on the Cisco ASR 1000 Series Router is an EIA/TIA-232 asynchronous, serial connection with no flow control and an RJ-45 connector. The console port is used to access the router and is located on the front panel of the Route Processor (RP).

For information on accessing the router using the console port, see the "Accessing the CLI Using a Directly-Connected Console" section on page 2-2.

Console Port Handling Overview

Users using the console port to access the router are automatically directed to the IOS command-line interface, by default.

If a user is trying to access the router through the console port and sends a break signal (a break signal can be sent by entering **Ctrl-C** or **Ctrl-Shift-6**, or by entering the **send break** command at the Telnet prompt) before connecting to the IOS command-line interface, the user is directed into a diagnostic mode by default if the nonRPIOS subpackages can be accessed.

These settings can be changed by configuring a transport map for the console port and applying that transport map to the console interface.

Telnet and SSH Overview for the Cisco ASR 1000 Series Routers

Telnet and Secure Shell (SSH) on the Cisco ASR 1000 Series Routers can be configured and handled like Telnet and SSH on other Cisco platforms. For information on traditional Telnet, see the **line** command in the *Cisco IOS Terminal Services Command Reference guide* located at: http://www.cisco.com/en/US/docs/ios/12_2/termserv/command/reference/trflosho.html#wp1029818.

For information on configuring traditional SSH, see the "Configuring Secure Shell" chapter of the *Cisco IOS Security Configuration Guide* located at: http://www.cisco.com/en/US/docs/ios/12_2/security/configuration/guide/scfssh.html.

The Cisco ASR 1000 Series Routers also introduces persistent Telnet and persistent SSH. Persistent Telnet and persistent SSH allow network administrators to more clearly define the treatment of incoming traffic when users access the router through the Management Ethernet port using Telnet or SSH. Notably, persistent Telnet and persistent SSH provide more robust network access by allowing the router to be configured to be accessible through the Ethernet Management port using Telnet or SSH even when the IOS process has failed.

Persistent Telnet and Persistent SSH Overview

In traditional Cisco routers, accessing the router using Telnet or SSH is not possible in the event of an IOS failure. When Cisco IOS fails on a traditional Cisco router, the only method of accessing the router is through the console port. Similarly, if all active IOS processes have failed on a Cisco ASR 1000 Series Router that is not using persistent Telnet or persistent SSH, the only method of accessing the router is through the console port.

With persistent Telnet and persistent SSH, however, users can configure a transport map that defines the treatment of incoming Telnet or SSH traffic on the Management Ethernet interface. Among the many configuration options, a transport map can be configured to direct all traffic to the IOS command-line interface, diagnostic mode, or to wait for an IOS vty line to become available and then direct users into diagnostic mode when the user sends a break signal while waiting for the IOS vty line to become available. If a user uses Telnet or SSH to access diagnostic mode, that Telnet or SSH connection will be usable even in scenarios when no IOS process is active. Therefore, persistent Telnet and persistent SSH introduce the ability to access the router via diagnostic mode when the IOS process is not active. For information on diagnostic mode, see the "Understanding the Diagnostic Mode" section on page 2-7.

See the "Configuring Persistent Telnet" section on page 3-5 and the "Configuring Persistent SSH" section on page 3-8 for information on the various other options that are configurable using persistent Telnet or persistent SSH transport maps.

Configuring a Console Port Transport Map

This task describes how to configure a transport map for a console port interface on the Cisco ASR 1000 Series Router.

SUMMARY STEPS

1. (Required) **enable**
2. (Required) **configure terminal**
3. (Required) **transport-map type console** *transport-map-name*
4. (Required) **connection wait** [**allow interruptible** | **none** {**disconnect**}]
5. (Optional) **banner** [**diagnostic** | **wait**] *banner-message*
6. (Required) **exit**
7. (Required) **transport type console** *console-line-number* **input** *transport-map-name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**transport-map type console** *transport-map-name* **Example:** `Router(config)# transport-map type console consolehandler`	Creates and names a transport map for handling console connections, and enter transport map configuration mode.
Step 4	**connection wait** [**allow interruptible** | **none**] **Example:** `Router(config-tmap)# connection wait none`	Specifies how a console connection will be handled using this transport map: • **allow interruptible**—The console connection waits for an IOS vty line to become available, and also allows user to enter diagnostic mode by interrupting a console connection waiting for the IOS vty line to become available. This is the default setting. **Note** Users can interrupt a waiting connection by entering **Ctrl-C** or **Ctrl-Shift-6**. • **none**—The console connection immediately enters diagnostic mode.

	Command or Action	Purpose

Step 5	**banner** [**diagnostic** | **wait**] *banner-message*
	
	Example:
	`Router(config-tmap)# banner diagnostic X`
	`Enter TEXT message.  End with the character 'X'.`
	`--Welcome to Diagnostic Mode--`
	`X`
	`Router(config-tmap)#`
		(Optional) Creates a banner message that will be seen by users entering diagnostic mode or waiting for the IOS vty line as a result of the console transport map configuration.
		• **diagnostic**—Creates a banner message seen by users directed into diagnostic mode as a result of the console transport map configuration.
		• **wait**—Creates a banner message seen by users waiting for the IOS vty to become available.
		• *banner-message*—The banner message, which begins and ends with the same delimiting character.

Step 6	**exit** **Example:** `Router(config-tmap)# exit`	Exits transport map configuration mode to re-enter global configuration mode.
Step 7	**transport type console** *console-line-number* **input** *transport-map-name* **Example:** `Router(config)# transport type console 0 input consolehandler`	Applies the settings defined in the transport map to the console interface. The *transport-map-name* for this command must match the *transport-map-name* defined in the **transport-map type console** command.

Examples

In the following example, a transport map to set console port access policies is created and attached to console port 0:

```
Router(config)# transport-map type console consolehandler
Router(config-tmap)# connection wait allow interruptible
Router(config-tmap)# banner diagnostic X
Enter TEXT message.  End with the character 'X'.
Welcome to diagnostic mode
X
Router(config-tmap)# banner wait X
Enter TEXT message.  End with the character 'X'.
Waiting for IOS vty line
X
Router(config-tmap)# exit

Router(config)# transport type console 0 input consolehandler
```

Configuring Persistent Telnet

This task describes how to configure persistent Telnet on the Cisco ASR 1000 Series Routers.

Prerequisites

For a persistent Telnet connection to access an IOS vty line on the Cisco ASR 1000 Series Router, local login authentication must be configured for the vty line (the **login** command in line configuration mode). If local login authentication is not configured, users will not be able to access IOS using a Telnet connection into the Management Ethernet interface with an applied transport map. Diagnostic mode will still be accessible in this scenario.

SUMMARY STEPS

1. (Required) **enable**
2. (Required) **configure terminal**
3. (Required) **transport-map type persistent telnet** *transport-map-name*
4. (Required) **connection wait** [**allow** {**interruptible**} | **none** {**disconnect**}]
5. (Optional) **banner** [**diagnostic** | **wait**] *banner-message*
6. (Required) **transport interface GigabitEthernet 0**
7. (Required) **exit**
8. (Required) **transport type persistent telnet input** *transport-map-name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**transport-map type persistent telnet** *transport-map-name* **Example:** `Router(config)# transport-map type persistent telnet telnethandler`	Creates and names a transport map for handling persistent Telnet connections, and enters transport map configuration mode.

	Command or Action	Purpose
Step 4	**connection wait** [**allow** {**interruptible**}| **none** {**disconnect**}] **Example:** `Router(config-tmap)# connection wait none`	Specifies how a persistent Telnet connection will be handled using this transport map: • **allow**—The Telnet connection waits for an IOS vty line to become available, and exits the router if interrupted. • **allow interruptible**—The Telnet connection waits for the IOS vty line to become available, and also allows user to enter diagnostic mode by interrupting a Telnet connection waiting for the IOS vty line to become available. This is the default setting. **Note** Users can interrupt a waiting connection by entering **Ctrl-C** or **Ctrl-Shift-6**. • **none**—The Telnet connection immediately enters diagnostic mode. • **none disconnect**—The Telnet connection does not wait for the IOS vty line and does not enter diagnostic mode, so all Telnet connections are rejected if no vty line is immediately available in IOS.

Step 5	**banner** [**diagnostic** | **wait**] *banner-message*
	
	Example:
	`Router(config-tmap)# banner diagnostic X`
	`Enter TEXT message.  End with the character 'X'.`
	`--Welcome to Diagnostic Mode--`
	`X`
	`Router(config-tmap)#`
		(Optional) Creates a banner message that will be seen by users entering diagnostic mode or waiting for the IOS vty line as a result of the persistent Telnet configuration.
		• **diagnostic**—creates a banner message seen by users directed into diagnostic mode as a result of the persistent Telnet configuration.
		• **wait**—creates a banner message seen by users waiting for the vty line to become available.
		• *banner-message*—the banner message, which begins and ends with the same delimiting character.

Step 6	**transport interface gigabitethernet 0** **Example:** `Router(config-tmap)# transport interface gigabitethernet 0`	Applies the transport map settings to the Management Ethernet interface (interface gigabitethernet 0). Persistent Telnet can only be applied to the Management Ethernet interface on the Cisco ASR 1000 Series Routers. This step must be taken before applying the transport map to the Management Ethernet interface.

	Command or Action	Purpose
Step 7	**exit** **Example:** `Router(config-tmap)# exit`	Exits transport map configuration mode to re-enter global configuration mode.
Step 8	**transport type persistent telnet input** *transport-map-name* **Example:** `Router(config)# transport type persistent telnet input telnethandler`	Applies the settings defined in the transport map to the Management Ethernet interface. The *transport-map-name* for this command must match the *transport-map-name* defined in the **transport-map type persistent telnet** command.

Examples

In the following example, a transport map that will make all Telnet connections wait for an IOS vty line to become available before connecting to the router, while also allowing the user to interrupt the process and enter diagnostic mode, is configured and applied to the Management Ethernet interface (interface gigabitethernet 0).

A diagnostic and a wait banner are also configured.

The transport map is then applied to the interface when the **transport type persistent telnet input** command is entered to enable persistent Telnet.

```
Router(config)# transport-map type persistent telnet telnethandler
Router(config-tmap)# connection wait allow interruptible
Router(config-tmap)# banner diagnostic X
Enter TEXT message.  End with the character 'X'.
--Welcome to Diagnostic Mode--
X
Router(config-tmap)# banner wait X
Enter TEXT message.  End with the character 'X'.
--Waiting for IOS Process--
X

Router(config-tmap)# transport interface gigabitethernet 0
Router(config-tmap)# exit
Router(config)# transport type persistent telnet input telnethandler
```

Configuring Persistent SSH

This task describes how to configure persistent SSH on the Cisco ASR 1000 Series Routers.

SUMMARY STEPS

1. (Required) **enable**
2. (Required) **configure terminal**
3. (Required) **transport-map type persistent ssh** *transport-map-name*
4. (Required) **connection wait** [**allow** {**interruptible**} | **none** {**disconnect**}]
5. (Required) **rsa keypair-name** *rsa-keypair-name*
6. (Optional) **authentication-retries** *number-of-retries*
7. (Optional) **banner** [**diagnostic** | **wait**] *banner-message*
8. (Optional) **time-out** *timeout-interval-in-seconds*
9. (Required) **transport interface GigabitEthernet 0**
10. (Required) **exit**
11. (Required) **transport type persistent ssh input** *transport-map-name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**transport-map type persistent ssh** *transport-map-name* **Example:** `Router(config)# transport-map type persistent ssh sshhandler`	Creates and names a transport map for handling persistent SSH connections, and enters transport map configuration mode.

	Command or Action	Purpose
Step 4	**connection wait** [**allow** {**interruptible**}| **none** {**disconnect**}] **Example:** `Router(config-tmap)# connection wait allow interruptible`	Specifies how a persistent SSH connection will be handled using this transport map: • **allow**—The SSH connection waits for the vty line to become available, and exits the router if interrupted. • **allow interruptible**—The SSH connection waits for the vty line to become available, and also allows users to enter diagnostic mode by interrupting a SSH connection waiting for the vty line to become available. This is the default setting. **Note** Users can interrupt a waiting connection by entering **Ctrl-C** or **Ctrl-Shift-6**. • **none**—The SSH connection immediately enters diagnostic mode. • **none disconnect**—The SSH connection does not wait for the vty line from IOS and does not enter diagnostic mode, so all SSH connections are rejected if no vty line is immediately available.
Step 5	**rsa keypair-name** *rsa-keypair-name* **Example:** `Router(config-tmap)# rsa keypair-name sshkeys`	Names the RSA keypair to be used for persistent SSH connections. For persistent SSH connections, the RSA keypair name must be defined using this command in transport map configuration mode. The RSA keypair definitions defined elsewhere on the router, such as through the use of the **ip ssh rsa keypair-name** command, do not apply to persistent SSH connections. No *rsa-keypair-name* is defined by default.
Step 6	**authentication-retries** *number-of-retries* **Example:** `Router(config-tmap)# authentication-retries 4`	(Optional) Specifies the number of authentication retries before dropping the connection. The default *number-of-retries* is 3.

Step 7	**banner** [**diagnostic** | **wait**] *banner-message*
	
	Example:
	`Router(config-tmap)# banner diagnostic X Enter TEXT message.  End with the character 'X'.`
	`--Welcome to Diagnostic Mode--`
	`X`
	`Router(config-tmap)#`
		(Optional) Creates a banner message that will be seen by users entering diagnostic mode or waiting for the vty line as a result of the persistent SSH configuration.
		• **diagnostic**—Creates a banner message seen by users directed into diagnostic mode as a result of the persistent SSH configuration.
		• **wait**—Creates a banner message seen by users waiting for the vty line to become active.
		• *banner-message*—The banner message, which begins and ends with the same delimiting character.

	Command or Action	Purpose
Step 8	**time-out** *timeout-interval* **Example:** `Router(config-tmap)# time-out 30`	(Optional) Specifies the SSH time-out interval in seconds. The default *timeout-interval* is 120 seconds.
Step 9	**transport interface gigabitethernet 0** **Example:** `Router(config-tmap)# transport interface gigabitethernet 0`	Applies the transport map settings to the Management Ethernet interface (interface gigabitethernet 0). Persistent SSH can only be applied to the Management Ethernet interface on the Cisco ASR 1000 Series Routers.
Step 10	**exit** **Example:** `Router(config-tmap)# exit`	Exits transport map configuration mode to re-enter global configuration mode.
Step 11	**transport type persistent ssh input** *transport-map-name* **Example:** `Router(config)# transport type persistent ssh input sshhandler`	Applies the settings defined in the transport map to the Management Ethernet interface. The *transport-map-name* for this command must match the *transport-map-name* defined in the **transport-map type persistent ssh** command.

Examples

In the following example, a transport map that will make all SSH connections wait for the vty line to become active before connecting to the router is configured and applied to the Management Ethernet interface (interface gigabitethernet 0). The RSA keypair is named sshkeys.

This example only uses the commands required to configure persistent SSH.

```
Router(config)# transport-map type persistent ssh sshhandler
Router(config-tmap)# connection wait allow
Router(config-tmap)# rsa keypair-name sshkeys
Router(config-tmap)# transport interface gigabitethernet 0
```

In the following example, a transport map is configured that will apply the following settings to any users attempting to access the Management Ethernet port via SSH:

- Users using SSH will wait for the vty line to become active, but will enter diagnostic mode if the attempt to access IOS through the vty line is interrupted.
- The RSA keypair name is sshkeys
- The connection allows one authentication retry.
- The banner "`--Welcome to Diagnostic Mode--`" will appear if diagnostic mode is entered as a result of SSH handling through this transport map.
- The banner "`--Waiting for vty line--`" will appear if the connection is waiting for the vty line to become active.

The transport map is then applied to the interface when the **transport type persistent ssh input** command is entered to enable persistent SSH:

```
Router(config)# transport-map type persistent ssh sshhandler
Router(config-tmap)# connection wait allow interruptible
```

```
Router(config-tmap)# rsa keypair-name sshkeys
Router(config-tmap)# authentication-retries 1
Router(config-tmap)# banner diagnostic X
Enter TEXT message.  End with the character 'X'.
--Welcome to Diagnostic Mode--
X
Router(config-tmap)#banner wait X
Enter TEXT message.  End with the character 'X'.
--Waiting for vty line--
X
Router(config-tmap)# time-out 30
Router(config-tmap)# transport interface gigabitethernet 0
Router(config-tmap)# exit

Router(config)# transport type persistent ssh input sshhandler
```

Viewing Console Port, SSH, and Telnet Handling Configurations

Use the **show transport-map** [**all** | **name** *transport-map-name* | **type** [**console** | **persistent** [**ssh** | **telnet**]]] EXEC or privileged EXEC command to view the transport map configurations.

In the following example, a console port, persistent SSH, and persistent Telnet transport are configured on the router and various forms of the **show transport-map** command are entered to illustrate the various ways the **show transport-map** command can be entered to gather transport map configuration information.

```
Router# show transport-map all
Transport Map:
  Name: consolehandler
  Type: Console Transport

Connection:
  Wait option: Wait Allow Interruptable
  Wait banner:

Waiting for the IOS CLI

  bshell banner:

Welcome to Diagnostic Mode


Transport Map:
  Name: sshhandler
  Type: Persistent SSH Transport

Interface:
  GigabitEthernet0

Connection:
  Wait option: Wait Allow Interruptable
  Wait banner:

Waiting for IOS prompt

  Bshell banner:

Welcome to Diagnostic Mode


SSH:
```

```
  Timeout: 120
  Authentication retries: 5
  RSA keypair: sshkeys

Transport Map:
  Name: telnethandler
  Type: Persistent Telnet Transport

Interface:
  GigabitEthernet0

Connection:
  Wait option: Wait Allow Interruptable
  Wait banner:

Waiting for IOS process

  Bshell banner:

Welcome to Diagnostic Mode


Transport Map:
  Name: telnethandling1
  Type: Persistent Telnet Transport

Connection:
  Wait option: Wait Allow


Router# show transport-map type console
Transport Map:
  Name: consolehandler
  Type: Console Transport

Connection:
  Wait option: Wait Allow Interruptable
  Wait banner:

Waiting for the IOS CLI

  Bshell banner:

Welcome to Diagnostic Mode


Router# show transport-map type persistent ssh
Transport Map:
  Name: sshhandler
  Type: Persistent SSH Transport

Interface:
  GigabitEthernet0

Connection:
  Wait option: Wait Allow Interruptable
  Wait banner:

Waiting for IOS prompt

  Bshell banner:

Welcome to Diagnostic Mode
```

```
SSH:
  Timeout: 120
  Authentication retries: 5
  RSA keypair: sshkeys

Router# show transport-map type persistent telnet
Transport Map:
  Name: telnethandler
  Type: Persistent Telnet Transport

Interface:
  GigabitEthernet0

Connection:
  Wait option: Wait Allow Interruptable
  Wait banner:

Waiting for IOS process

  Bshell banner:

Welcome to Diagnostic Mode


Transport Map:
  Name: telnethandling1
  Type: Persistent Telnet Transport

Connection:
  Wait option: Wait Allow


Router# show transport-map name telnethandler
Transport Map:
  Name: telnethandler
  Type: Persistent Telnet Transport

Interface:
  GigabitEthernet0

Connection:
  Wait option: Wait Allow Interruptable
  Wait banner:

Waiting for IOS process

  Bshell banner:

Welcome to Diagnostic Mode


Router# show transport-map name consolehandler
Transport Map:
  Name: consolehandler
  Type: Console Transport

Connection:
  Wait option: Wait Allow Interruptable
  Wait banner:

Waiting for the IOS CLI

  Bshell banner:
```

```
Welcome to Diagnostic Mode


Router# show transport-map name sshhandler
Transport Map:
  Name: sshhandler
  Type: Persistent SSH Transport

Interface:
  GigabitEthernet0

Connection:
  Wait option: Wait Allow Interruptable
  Wait banner:

Waiting for IOS prompt

  Bshell banner:

Welcome to Diagnostic Mode


SSH:
  Timeout: 120
  Authentication retries: 5
  RSA keypair: sshkeys

Router#
```

The **show platform software configuration access policy** command can be used to view the current configurations for the handling of incoming console port, SSH, and Telnet connections. The output of this command provides the current wait policy for each type of connection, as well as any information on the currently configured banners. Unlike **show transport-map**, this command is available in diagnostic mode so it can be entered in cases when you need transport map configuration information but cannot access the IOS CLI.

```
Router# show platform software configuration access policy
The current access-policies

Method      : telnet
Rule        : wait
Shell banner:
Wait banner :

Method      : ssh
Rule        : wait
Shell banner:
Wait banner :

Method      : console
Rule        : wait with interrupt
Shell banner:
Wait banner :
```

In the following example, the connection policy and banners are set for a persistent SSH transport map, and the transport map is enabled.

The **show platform software configuration access policy** output is given both before the new transport map is enabled and after the transport map is enabled so the changes to the SSH configuration are illustrated in the output.

```
Router# show platform software configuration access policy
```

```
The current access-policies

Method      : telnet
Rule        : wait with interrupt
Shell banner:
Welcome to Diagnostic Mode

Wait banner :
Waiting for IOS Process


Method      : ssh
Rule        : wait
Shell banner:
Wait banner :

Method      : console
Rule        : wait with interrupt
Shell banner:
Wait banner :


Router# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.

Router(config)# transport-map type persistent ssh sshhandler
Router(config-tmap)# connection wait allow interruptible
Router(config-tmap)# banner diagnostic X
Enter TEXT message.  End with the character 'X'.
Welcome to Diag Mode
X
Router(config-tmap)# banner wait X
Enter TEXT message.  End with the character 'X'.
Waiting for IOS
X
Router(config-tmap)# rsa keypair-name sshkeys
Router(config-tmap)# transport interface gigabitethernet 0
Router(config-tmap)# exit

Router(config)# transport type persistent ssh input sshhandler
Router(config)# exit

Router# show platform software configuration access policy
The current access-policies

Method      : telnet
Rule        : wait with interrupt
Shell banner:
Welcome to Diagnostic Mode

Wait banner :
Waiting for IOS process


Method      : ssh
Rule        : wait with interrupt
Shell banner:
Welcome to Diag Mode

Wait banner :
Waiting for IOS


Method      : console
```

```
Rule        : wait with interrupt
Shell banner:
Wait banner :
```

Important Notes and Restrictions

The important notes and restriction pertaining to the console port, SSH, and telnet handling include:

- The Telnet and SSH settings made in the transport map override any other Telnet or SSH settings when the transport map is applied to the Management Ethernet interface.
- Only local usernames and passwords can be used to authenticate users entering a Management Ethernet interface. AAA authentication is not available for users accessing the router through a Management Ethernet interface using persistent Telnet or persistent SSH.
- Applying a transport map to a Management Ethernet interface with active Telnet or SSH sessions can disconnect the active sessions. Removing a transport map from an interface, however, does not disconnect any active Telnet or SSH sessions.
- Configuring the diagnostic and wait banners is optional but recommended. The banners are especially useful as indicators to users of the status of their Telnet or SSH attempts.

Consolidated Packages and SubPackages Management

This chapter discusses how consolidated packages and software subpackages (individual and optional) are managed and are used to run the Cisco ASR 1000 Series Aggregation Services Routers. It contains the following sections:

- Running the Cisco ASR 1000 Series Routers: An Overview, page 4-1
- Software File Management Using Command Sets, page 4-4
- Managing and Configuring the Router to Run Using Consolidated Packages and Individual SubPackages, page 4-6
- Upgrading Individual SubPackages, page 4-18

Note This section of the document does not discuss limited downtime upgrades. For information on limited downtime upgrades, see the "Software Upgrade Processes Supported by Cisco ASR 1000 Series Routers" section on page 5-1.

Running the Cisco ASR 1000 Series Routers: An Overview

The Cisco ASR 1000 Series Routers can be run using a complete consolidated package or using individual subpackages.

This section covers the following topics:

- Running the Cisco ASR 1000 Series Routers Using Individual and Optional SubPackages: An Overview, page 4-2
- Running the Cisco ASR 1000 Series Routers Using a Consolidated Package: An Overview, page 4-2
- Running the Cisco ASR 1000 Series Routers: A Summary, page 4-3

Running the Cisco ASR 1000 Series Routers Using Individual and Optional SubPackages: An Overview

The Cisco ASR 1000 Series Routers can be configured to run using individual subpackages and optional subpackages.

When the router is configured to run using individual and optional subpackages:

- Each individual subpackage within a consolidated package is extracted onto the router as its own file.
- Additionally, any optional subpackages must be separately downloaded and stored in the same directory with the provisioning file and the other individual subpackages that have been extracted.
- The router then runs by accessing each file as needed for operational purposes. All individual and optional subpackage files must be stored in the same directory on the router for the router to run properly using individual subpackages.

When a Cisco ASR 1000 Series Router is configured to run using individual and optional subpackages, software image content from the RP is copied into memory on an as-needed basis only. Memory is conserved for other router processes, such as passing traffic. The router, therefore, allows for the highest peak traffic load when configured to run using individual subpackages.

When running the Cisco ASR 1000 Series Routers using individual and optional subpackages, the router must be configured to boot using the provisioning file that was included in the consolidated package with the individual subpackage files. This provisioning file must also be in the same directory as the individual and optional subpackage files. The router boots faster when configured to run using individual and optional subpackages than it does when configured to run using a consolidated package.

A Cisco ASR 1000 Series Router cannot be configured to run individual and optional subpackages stored on a TFTP or any other network server. The individual and optional subpackages along with the provisioning file must be copied onto the bootflash: file system to run the router using this method.

Running the Cisco ASR 1000 Series Routers Using a Consolidated Package: An Overview

The Cisco ASR 1000 Series Routers can also be configured to run using a consolidated package.

Note Booting the router from a consolidated package is not supported for installation of optional subpackages. For more information, see the "Running the Cisco ASR 1000 Series Routers Using Individual and Optional SubPackages: An Overview" section on page 4-2.

When the router is configured to run using a consolidated package, the entire consolidated package file is copied onto the router or accessed by the router via TFTP or another network transport method. The router runs using the consolidated package file.

When a Cisco ASR 1000 Series Router is configured to run using the consolidated package file, more memory is required to process router requests because the router has to search one larger file for every request. The peak amount of memory available for passing network traffic is therefore lower when the router is configured to run using a consolidated package.

A Cisco ASR 1000 Series Router configured to run using a consolidated package is booted by booting the consolidated package file. Because this file is large, the boot process for routers running using the consolidated package is slower than the boot process for routers running individual subpackages.

A Cisco ASR 1000 Series Router configured to run using a consolidated package does have some advantages over a Cisco ASR 1000 Series Router configured to run individual subpackages. First, a consolidated package can be booted and utilized using TFTP or another network transport method. Secondly, configuring the router to use the one consolidated package file is easier than managing several individual subpackage files. Running the router using a consolidated package may be the right method of running the router in certain networking environments.

The consolidated package should be stored on bootflash:, usb[0-1]:, or a remote file system when this method is used to run the router.

Running the Cisco ASR 1000 Series Routers: A Summary

This section summarizes the advantages and disadvantages of each method of running your Cisco ASR 1000 Series Routers.

The advantages of running your router using individual subpackages include:

- Optimizes router memory usage—When the router is booting using the individual and optional subpackage method, software image content from the Router Processor (RP) is copied into memory on an as-needed basis only, which conserves memory for other router processes. The router boots fastest and allows for the highest peak traffic load when booted using the individual subpackage boot approach.

The advantages of running your router using a consolidated package include:

- Simplified installation—Only one software file needs to be managed instead of several separate images.
- Storage—A consolidated package, unlike individual subpackages, can be used to run the router while being stored in bootflash:, on a USB Flash disk, or on a network server. A consolidated package can be booted and utilized using TFTP or another network transport method, while the individual subpackage method requires the individual subpackage files to be copied into the bootflash: file directory on the router.

Table 4-1 summarizes the advantages and disadvantages of the approaches.

Table 4-1 Advantages and Disadvantages of Running Individual subPackages and Consolidated Packages

Approach	Advantages	Disadvantages
Individual and optional subpackages **Note** This method is required if you need to install any optional subpackages for your system.	• Optimizes memory usage, allowing for faster boot times and increased traffic loads.	• Multiple software subpackages more difficult to manage. • Cannot be booted from TFTP or any other network server. If you are going to use the individual subpackage boot method, each individual subpackage file must be placed in the bootflash: directory. • Individual and optional subpackage files and the provisioning file must be stored in bootflash:.
Consolidated Package	• Easier management. Only have to manage one file instead of many files. • A consolidated package file can be stored in bootflash:, on a USB Flash disk, or on any TFTP or other network server.	• Slower boot times and lessened maximum system scalability because the larger image must be processed at all times.

Software File Management Using Command Sets

Software files can be managed on the Cisco ASR 1000 Series Routers using three distinct command sets. This section provides overviews of the following command sets:

- The request platform Command Set, page 4-4
- The copy Command, page 4-5
- The issu Command Set, page 4-5

The request platform Command Set

The **request platform software package** command is part of the larger **request platform** command set being introduced on the Cisco ASR 1000 Series Routers. For additional information on each **request platform** command and the options available with each command, see the *Cisco IOS Configuration Fundamentals Command Reference*.

The **request platform software package** command, which can be used to upgrade individual subpackages and a complete consolidated package, is used to upgrade software on the Cisco ASR 1000 Series Routers. Notably, the **request platform software package** command is the recommended way of performing an individual subpackage upgrade, and also provides the only method of no-downtime upgrades of individual subpackages on the router when the router is running individual subpackages.

The **request platform software package** command requires that the destination device or process be specified in the command line, so the commands can be used to upgrade software on both an active or a standby processor. The **request platform software package** command allows for no downtime software upgrades in many scenarios.

The basic syntax of the command is **request platform software package install rp** *rp-slot-number* **file** *file-URL*, where *rp-slot-number* is the number of the RP slot and *file-URL* is the path to the file being used to upgrade the Cisco ASR 1000 Series Router. The command has other options; see the **request platform software package** command references for information on all of the options available with this command set.

In Cisco IOS XE Release 2.4, the **request platform software package uninstall** command is introduced on the Cisco ASR 1000 Series Routers to remove provisioning of an optional subpackage from an RP. The syntax for the command follows the install form of the command: **request platform software package uninstall rp** *rp-slot-number* **type** *pkg*, where *rp-slot-number* is the number of the RP slot and *pkg* is the type of optional subpackage file to be removed.

Note For additional information on limited downtime upgrades using the **request platform software package** command set, see the "Software Upgrade Processes Supported by Cisco ASR 1000 Series Routers" section on page 5-1.

The copy Command

The **copy** command can be used to move consolidated packages and individual subpackages onto the router, though using this command to move individual subpackage files from one storage area to another is often inefficient (in these scenarios, it is almost always preferable to move the consolidated package, then extract the subpackages, or to extract the subpackages without moving the consolidated package).

To upgrade a consolidated package on the Cisco ASR 1000 Series Routers, copy the consolidated package onto a file system, usually bootflash: or usb[0-1]: on the router, using the **copy** command as you would on most other Cisco routers. After making this copy, configure the router to boot using the consolidated package file.

To upgrade the router and reboot using individual subpackages, copy the consolidated package onto the router using the **copy** command, enter the **request platform software package expand** command to extract the individual subpackages, and configure the router to boot using subpackages. Other methods, such as copying each individual subpackage in the same consolidated package from a directory or using the **request platform software package** command to extract the subpackages onto a router directory are also usable, though copying individual subpackages is often inefficient.

See the **copy** command reference for a list of the options that are available with the **copy** command.

The issu Command Set

The **issu** command set that is available on other Cisco routers is also available on the Cisco ASR 1000 Series Routers. The **issu** command set can be used to perform both consolidated package and individual subpackage upgrades.

The **issu** command set includes the **issu** command with the **loadversion**, **runversion**, **commitversion**, and **abortversion** options. See the **issu** command references for information on all of the options available with each of these commands.

A confirm prompt is displayed when the command **issu abortversion** is issued and when the standby has not reached the terminal (hot) state.

The following example shows a confirm prompt notification:

```
The system is without a fully initialized peer and service impact will occur. Proceed with
abort? [confirm]
```

The **issu runversion** command is optional for completing a consolidated package upgrade on the Cisco ASR 1000 Series Routers. You can enter **issu commitversion** from the active RP to complete the software upgrade, then use the **redundancy force-switchover** command to switch to the upgraded RP on this platform, if desired.

Note For additional information on limited downtime upgrades using the **issu** command set, see the "Software Upgrade Processes Supported by Cisco ASR 1000 Series Routers" section on page 5-1.

Managing and Configuring the Router to Run Using Consolidated Packages and Individual SubPackages

This section discusses the following topics:

- Quick Start Software Upgrade, page 4-6
- Managing and Configuring a Router to Run Using a Consolidated Package, page 4-7
- Managing and Configuring a Router to Run Using Individual SubPackages From a Consolidated Package, page 4-10
- Managing and Configuring a Router to Run Using Optional SubPackages, page 4-14
- Troubleshooting Software Mismatch with ESP Board ASR1000-ESP10-N, page 4-18

Quick Start Software Upgrade

The following instructions provide a quick start version of upgrading the software running the Cisco ASR 1000 Series Routers. These instructions assume you have access to the consolidated package and that the files will be stored in a bootflash: file system that is not storing any previously installed subpackages or consolidated packages and that has enough room for the file or files.

For more detailed installation examples, see the other sections of this chapter.

To upgrade the software using a quick start version, perform the following steps:

Step 1 Copy the consolidated package into bootflash: using the **copy** *URL-to-image* **bootflash:** command.

Step 2 If you want to run the router using individual subpackages, enter the **request platform software package expand file bootflash:***image-name* command. If you want to run the router using a consolidated package, skip this step.

Step 3 Enter the **dir bootflash:** command to verify your consolidated package or your extracted subpackages are in the directory.

Step 4 If you are trying to run individual subpackages, use the **delete bootdisk:***image-name* to delete the consolidated package. If you want to run the router using the consolidated package, skip this step.

Step 5 Set up the boot parameters for your boot. Set the configuration register to 0x2 by entering the **config-register 0x2102** global configuration command, and enter the **boot system flash bootflash:***image-name* (if running using the consolidated package) or **boot system flash bootflash:***provisionging-file-name* (if running using individual subpackages) global configuration command.

Step 6 Enter **copy running-config startup-config** to save your configuration.

Step 7 Enter the **reload** command to reload the router and finish the boot. The upgraded software should be running when the reload completes.

Managing and Configuring a Router to Run Using a Consolidated Package

Note Do not use this procedure if you also need to install any optional subpackages.

This section documents the following procedures:

- Managing and Configuring a Consolidated Package Using the copy Command, page 4-7
- Managing and Configuring a Consolidated Package Using the request platform software package install Command, page 4-8

Managing and Configuring a Consolidated Package Using the copy Command

To upgrade a consolidated package on the Cisco ASR 1000 Series Routers using the **copy** command, copy the consolidated package into the bootflash: directory on the router using the **copy** command as you would on most other Cisco routers. After making this copy, configure the router to boot using the consolidated package file.

In the following example, the consolidated package file is copied onto the bootflash: file system from TFTP. The config-register is then set to boot using **boot system** commands, and the **boot system** commands instruct the router to boot using the consolidated package stored in the bootflash: file system. The new configuration is then saved using the **copy running-config startup-config** command, and the system is then reloaded to complete the process.

```
Router# dir bootflash:
Directory of bootflash:/

   11  drwx       16384   Dec 4 2007 04:32:46 -08:00  lost+found
86401  drwx        4096   Dec 4 2007 06:06:24 -08:00  .ssh
14401  drwx        4096   Dec 4 2007 06:06:36 -08:00  .rollback_timer
28801  drwx        4096  Mar 18 2008 17:31:17 -07:00  .prst_sync
43201  drwx        4096   Dec 4 2007 04:34:45 -08:00  .installer
   13  -rw-       45977   Apr 9 2008 16:48:46 -07:00  target_support_output.tgz.tgz

928862208 bytes total (712273920 bytes free)
Router# copy tftp bootflash:
Address or name of remote host []? 172.17.16.81
Source filename []?
/auto/tftp-users/user/asr1000rp1-adventerprisek9.02.01.00.122-33.XNA.bin
Destination filename [asr1000rp1-adventerprisek9.02.01.00.122-33.XNA.bin]?
Accessing
tftp://172.17.16.81//auto/tftp-users/user/asr1000rp1-adventerprisek9.02.01.00.122-33.XNA.b
in...
```

```
Loading /auto/tftp-users/user/asr1000rp1-adventerprisek9.02.01.00.122-33.XNA.bin from
172.17.16.81 (via GigabitEthernet0):
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!
[OK - 208904396 bytes]

208904396 bytes copied in 330.453 secs (632176 bytes/sec)
Router# dir bootflash:
Directory of bootflash:/

   11  drwx       16384   Dec 4 2007 04:32:46 -08:00  lost+found
86401  drwx        4096   Dec 4 2007 06:06:24 -08:00  .ssh
14401  drwx        4096   Dec 4 2007 06:06:36 -08:00  .rollback_timer
28801  drwx        4096  Mar 18 2008 17:31:17 -07:00  .prst_sync
43201  drwx        4096   Dec 4 2007 04:34:45 -08:00  .installer
   12  -rw-   208904396  May 28 2008 16:17:34 -07:00
asr1000rp1-adventerprisek9.02.01.00.122-33.XNA.bin
   13  -rw-       45977   Apr 9 2008 16:48:46 -07:00  target_support_output.tgz.tgz

928862208 bytes total (503156736 bytes free)
Router# config t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#boot system flash
bootflash:asr1000rp1-adventerprisek9.02.01.00.122-33.XNA.bin
Router(config)#config-reg 0x2102
Router(config)#exit
Router#show run | include boot
boot-start-marker
boot system flash bootflash:asr1000rp1-adventerprisek9.02.01.00.122-33.XNA.bin
boot-end-marker
Router# copy run start
Destination filename [startup-config]?
Building configuration...
[OK]
Router# reload
```

Managing and Configuring a Consolidated Package Using the request platform software package install Command

In the following example, the **request platform software package install** command is used to upgrade a consolidated package running on RP 0. The **force** option, which forces the upgrade past any prompt (such as already having the same consolidated package installed), is used in this example.

```
Router# request platform software package install rp 0 file
bootflash:asr1000rp1-adventerprisek9.02.01.00.122-33.XNA.bin force
--- Starting installation state synchronization ---
Finished installation state synchronization

--- Starting file path checking ---
Finished file path checking

--- Starting image file verification ---
Checking image file names
Verifying image file locations
```

```
Locating image files and validating name syntax
Inspecting image file types

Processing image file constraints

Extracting super package content
Verifying parameters
Validating package type

Copying package files

Checking and verifying packages contained in super package
Creating candidate provisioning file

  WARNING:
  WARNING: Candidate software will be installed upon reboot
  WARNING:

Finished image file verification

--- Starting candidate package set construction ---
Verifying existing software set
Processing candidate provisioning file

Constructing working set for candidate package set
Constructing working set for running package set
Checking command output
Constructing merge of running and candidate packages
Finished candidate package set construction

--- Starting compatibility testing ---
Determining whether candidate package set is compatible

WARNING:
WARNING: Candidate software combination not found in compatibility database
WARNING:


Determining whether installation is valid
Determining whether installation is valid ... skipped
Checking IPC compatibility with running software
Checking IPC compatibility with running software ... skipped
Checking candidate package set infrastructure compatibility
Checking infrastructure compatibility with running software
Checking infrastructure compatibility with running software ... skipped
Finished compatibility testing

--- Starting commit of software changes ---
Updating provisioning rollback files
Creating pending provisioning file
Committing provisioning file
Finished commit of software changes

SUCCESS: Software provisioned.  New software will load on reboot.

Router# reload
```

Note A reload must be performed to finish this procedure.
The "Managing and Configuring a Consolidated Package Using the copy Command" section on page 4-7 includes an example of how to configure the router to boot using the consolidated package, and then an example of what happens after the reload is performed to finish the installation.

Managing and Configuring a Router to Run Using Individual SubPackages From a Consolidated Package

To run the router using individual subpackages from a consolidated package, follow one of the following procedures:

- Extracting a Consolidated Package and Booting Using the Provisioning File, page 4-10
- Copying a Set of Individual SubPackage Files, and Booting Using a Provisioning File, page 4-14

Extracting a Consolidated Package and Booting Using the Provisioning File

To extract a consolidated package and to boot using provisioning file, perform the following steps:

Step 1 Perform one of the following tasks:

- Copy the consolidated package file (or, in cases where you have every individual subpackage and a provisioning file for the subpackages available, each individual subpackage and the provisioning file) onto the bootflash: file system using the **copy** command. Make sure to copy the consolidated package into the bootflash: file system and directory where you want to store the provisioning file and the individual image subpackages. Enter the **request platform software package expand file bootflash:***url-to-Cisco-IOS-XE-imagename* command with no other option to extract the provisioning file and the individual subpackages out of the consolidated package file and into the current directory in bootflash:.
- Copy the consolidated package file onto any file system on your router, then enter the **request platform software package expand file** *file-system***:***url-to-Cisco-IOS-XE-imagename* **to bootflash:** command to extract the provisioning file and the individual image subpackages onto the bootflash: file system.

Note After performing this step, do not move any of the files. The bootup process cannot function properly unless all of the subpackages and the provisioning file are located in the same directory.
Also, do not rename the subpackage files. Only the provisioning file can be renamed, and the renaming of the provisioning file, if desired, should be done at this step before the router is rebooted.

Step 2 Configure the router to boot using the provisioning file.

The sequence below provides an example that would boot the router using the provisioning file named packages.conf that was stored with the other subpackages in the bootflash: file system:

```
Router(config)# no boot system
Router(config)# config-register 0x2102
Router(config)# boot system flash bootflash:packages.conf
Router(config)# exit
*May 11 01:31:04.815: %SYS-5-CONFIG_I: Configured from console by con
Router# copy running-config startup-config
Building configuration...
[OK]
Router# reload
```

Extracting the SubPackages and the Provisioning File: Example 1

The following example shows how to extract the individual subpackages and the provisioning file from a consolidated package that has already been placed in the directory where you want to store the individual subpackages and the provisioning file.

Output of the directory before and after the extraction is given to confirm the files were extracted.

```
Router# dir bootflash:
Directory of bootflash:/

   11  drwx       16384   Dec 4 2007 04:32:46 -08:00  lost+found
86401  drwx        4096   Dec 4 2007 06:06:24 -08:00  .ssh
14401  drwx        4096   Dec 4 2007 06:06:36 -08:00  .rollback_timer
28801  drwx        4096  Mar 18 2008 17:31:17 -07:00  .prst_sync
43201  drwx        4096   Dec 4 2007 04:34:45 -08:00  .installer
   12  -rw-   208904396   May 9 2008 14:36:31 -07:00
asr1000rp1-adventerprisek9.02.01.00.122-33.XNA.bin
   13  -rw-       45977   Apr 9 2008 16:48:46 -07:00  target_support_output.tgz.tgz

928862208 bytes total (503156736 bytes free)

Router# request platform software package expand file
bootflash:asr1000rp1-adventerprisek9.02.01.00.122-33.XNA.bin
Verifying parameters
Validating package type
Copying package files

SUCCESS: Finished expanding all-in-one software package.

Router# dir bootflash:
Directory of bootflash:/

   11  drwx       16384   Dec 4 2007 04:32:46 -08:00  lost+found
86401  drwx        4096   Dec 4 2007 06:06:24 -08:00  .ssh
14401  drwx        4096   Dec 4 2007 06:06:36 -08:00  .rollback_timer
28801  drwx        4096  Mar 18 2008 17:31:17 -07:00  .prst_sync
43201  drwx        4096   Dec 4 2007 04:34:45 -08:00  .installer
   12  -rw-   208904396   May 9 2008 14:36:31 -07:00
asr1000rp1-adventerprisek9.02.01.00.122-33.XNA.bin
57611  -rw-    47071436  May 22 2008 11:26:23 -07:00
asr1000rp1-espbase.02.01.00.122-33.XNA.pkg
57602  -rw-        5740  May 22 2008 11:26:22 -07:00
asr1000rp1-packages-adventerprisek9.02.01.00.122-33.XNA.conf
57612  -rw-    20334796  May 22 2008 11:26:24 -07:00
asr1000rp1-rpaccess.02.01.00.122-33.XNA.pkg
57613  -rw-    22294732  May 22 2008 11:26:24 -07:00
asr1000rp1-rpbase.02.01.00.122-33.XNA.pkg
57614  -rw-    21946572  May 22 2008 11:26:25 -07:00
asr1000rp1-rpcontrol.02.01.00.122-33.XNA.pkg
57615  -rw-    48099532  May 22 2008 11:26:26 -07:00
asr1000rp1-rpios-adventerprisek9.02.01.00.122-33.XNA.pkg
57616  -rw-    34324684  May 22 2008 11:26:27 -07:00
asr1000rp1-sipbase.02.01.00.122-33.XNA.pkg
57617  -rw-    22124748  May 22 2008 11:26:28 -07:00
asr1000rp1-sipspa.02.01.00.122-33.XNA.pkg
57603  -rw-        6256  May 22 2008 11:26:28 -07:00  packages.conf
   13  -rw-       45977   Apr 9 2008 16:48:46 -07:00  target_support_output.tgz.tgz

928862208 bytes total (286662656 bytes free)
```

Extracting the SubPackages and the Provisioning File: Example 2

The following example shows how to extract the individual subpackages and the provisioning file from a consolidated package that has already been placed on the router in a directory that will not store the individual subpackages and the provisioning file. In this particular example, the contents of a consolidated package stored in usb0: are extracted into bootflash:.

Output of the bootflash: directory before and after the extraction is given to confirm the files were extracted:

```
Router# dir usb0:
Directory of usb0:/

 1240  -rwx   208904396  May 27 2008 14:10:20 -07:00
asr1000rp1-adventerprisek9.02.01.00.122-33.XNA.bin

255497216 bytes total (40190464 bytes free)

Router# dir bootflash:
Directory of bootflash:/

   11  drwx       16384  Dec 13 2004 03:45:47 -08:00  lost+found
87937  drwx        4096  Dec 13 2004 03:46:45 -08:00  .rollback_timer
14657  drwx        4096  Dec 13 2004 03:47:17 -08:00  .installer
29313  drwx        4096  Dec 13 2004 03:53:00 -08:00  .ssh
   12  -rw-    33554432  Dec 13 2004 03:53:49 -08:00  nvram_00100
58625  drwx        4096   May 7 2008 17:27:51 -07:00  .prst_sync

945377280 bytes total (695246848 bytes free)

Router# request platform software package expand file
usb0:asr1000rp1-adventerprisek9.02.01.00.122-33.XNA.bin to bootflash:
Verifying parameters
Validating package type
Copying package files
SUCCESS: Finished expanding all-in-one software package.

Router# dir bootflash:
Directory of bootflash:/

   11  drwx       16384  Dec 13 2004 03:45:47 -08:00  lost+found
87937  drwx        4096  Dec 13 2004 03:46:45 -08:00  .rollback_timer
14657  drwx        4096  Dec 13 2004 03:47:17 -08:00  .installer
29313  drwx        4096  Dec 13 2004 03:53:00 -08:00  .ssh
   12  -rw-    33554432  Dec 13 2004 03:53:49 -08:00  nvram_00100
43979  -rw-    47071436  May 27 2008 14:25:01 -07:00
asr1000rp1-espbase.02.01.00.122-33.XNA.pkg
58625  drwx        4096   May 7 2008 17:27:51 -07:00  .prst_sync
43970  -rw-        5740  May 27 2008 14:25:00 -07:00
asr1000rp1-packages-adventerprisek9.02.01.00.122-33.XNA.conf
43980  -rw-    20334796  May 27 2008 14:25:01 -07:00
asr1000rp1-rpaccess.02.01.00.122-33.XNA.pkg
43981  -rw-    22294732  May 27 2008 14:25:02 -07:00
asr1000rp1-rpbase.02.01.00.122-33.XNA.pkg
43982  -rw-    21946572  May 27 2008 14:25:03 -07:00
asr1000rp1-rpcontrol.02.01.00.122-33.XNA.pkg
43983  -rw-    48099532  May 27 2008 14:25:04 -07:00
asr1000rp1-rpios-adventerprisek9.02.01.00.122-33.XNA.pkg
43984  -rw-    34324684  May 27 2008 14:25:05 -07:00
asr1000rp1-sipbase.02.01.00.122-33.XNA.pkg
43985  -rw-    22124748  May 27 2008 14:25:05 -07:00
asr1000rp1-sipspa.02.01.00.122-33.XNA.pkg
43971  -rw-        6256  May 27 2008 14:25:05 -07:00  packages.conf
```

```
945377280 bytes total (478752768 bytes free)
```

Extracting the SubPackages, Configuring the Router to Boot Using the Provisioning File, and Reloading the Router: Example 3

In the following example, the provisioning file and the individual subpackages are extracted from a consolidated package. The router is then configured to boot using the provisioning file. This example also shows the config-register being set and the running configuration being saved because these tasks must be performed for the router to reload properly. The router is then reloaded to complete the process.

```
Router# dir bootflash:
Directory of bootflash:/

   11  drwx       16384   Dec 4 2007 04:32:46 -08:00  lost+found
86401  drwx        4096   Dec 4 2007 06:06:24 -08:00  .ssh
14401  drwx        4096   Dec 4 2007 06:06:36 -08:00  .rollback_timer
28801  drwx        4096  Mar 18 2008 17:31:17 -07:00  .prst_sync
43201  drwx        4096   Dec 4 2007 04:34:45 -08:00  .installer
   12  -rw-   208904396   May 9 2008 14:36:31 -07:00
asr1000rp1-adventerprisek9.02.01.00.122-33.XNA.bin
   13  -rw-       45977   Apr 9 2008 16:48:46 -07:00  target_support_output.tgz.tgz

928862208 bytes total (503156736 bytes free)

Router# request platform software package expand file
bootflash:asr1000rp1-adventerprisek9.02.01.00.122-33.XNA.bin
Verifying parameters
Validating package type
Copying package files

SUCCESS: Finished expanding all-in-one software package.

Router# dir bootflash:
Directory of bootflash:/

   11  drwx       16384   Dec 4 2007 04:32:46 -08:00  lost+found
86401  drwx        4096   Dec 4 2007 06:06:24 -08:00  .ssh
14401  drwx        4096   Dec 4 2007 06:06:36 -08:00  .rollback_timer
28801  drwx        4096  Mar 18 2008 17:31:17 -07:00  .prst_sync
43201  drwx        4096   Dec 4 2007 04:34:45 -08:00  .installer
   12  -rw-   208904396   May 9 2008 14:36:31 -07:00
asr1000rp1-adventerprisek9.02.01.00.122-33.XNA.bin
57611  -rw-    47071436  May 22 2008 11:26:23 -07:00
asr1000rp1-espbase.02.01.00.122-33.XNA.pkg
57602  -rw-        5740  May 22 2008 11:26:22 -07:00
asr1000rp1-packages-adventerprisek9.02.01.00.122-33.XNA.conf
57612  -rw-    20334796  May 22 2008 11:26:24 -07:00
asr1000rp1-rpaccess.02.01.00.122-33.XNA.pkg
57613  -rw-    22294732  May 22 2008 11:26:24 -07:00
asr1000rp1-rpbase.02.01.00.122-33.XNA.pkg
57614  -rw-    21946572  May 22 2008 11:26:25 -07:00
asr1000rp1-rpcontrol.02.01.00.122-33.XNA.pkg
57615  -rw-    48099532  May 22 2008 11:26:26 -07:00
asr1000rp1-rpios-adventerprisek9.02.01.00.122-33.XNA.pkg
57616  -rw-    34324684  May 22 2008 11:26:27 -07:00
asr1000rp1-sipbase.02.01.00.122-33.XNA.pkg
57617  -rw-    22124748  May 22 2008 11:26:28 -07:00
asr1000rp1-sipspa.02.01.00.122-33.XNA.pkg
57603  -rw-        6256  May 22 2008 11:26:28 -07:00  packages.conf
   13  -rw-       45977   Apr 9 2008 16:48:46 -07:00  target_support_output.tgz.tgz

928862208 bytes total (286662656 bytes free)
```

```
Router(config)# no boot system
Router(config)# config-register 0x2102
Router(config)# boot system flash bootflash:packages.conf
Router(config)# exit
Router# copy run start
Router# reload
```

Copying a Set of Individual SubPackage Files, and Booting Using a Provisioning File

To copy a set of individual subpackage files and to boot using a provisioning file, perform the following steps:

Note Although this upgrade method works, it is less efficient than other methods of upgrading the router's software.

Step 1 Copy each individual subpackage and the provisioning file into the bootflash: directory using the **copy** command. Note that this method of running the router will only work if all the individual subpackages for a release and a provisioning file are downloaded onto the router and stored in the bootflash: directory. No other file directories should be used for booting the router using individual subpackages.

The files can also be moved on the router physically using a USB Flash drive.

Step 2 Configure the router to boot using the provisioning file.

The sequence below provides an example that describes how to boot the router using the provisioning file named packages.conf that was stored with the other subpackages in the bootflash: file system. The router runs using individual subpackages once the reload is complete.

```
Router(config)# no boot system
Router(config)# config-register 0x2102
Router(config)# boot system flash bootflash:packages.conf
Router(config)# exit
*May 11 01:31:04.815: %SYS-5-CONFIG_I: Configured from console by con
Router# write memory
Building configuration...
[OK]
Router# reload
```

Managing and Configuring a Router to Run Using Optional SubPackages

To manage and configure a Cisco ASR 1000 Series Router to run using optional subpackages, perform the following tasks:

- Installing an Optional SubPackage, page 4-15
- Uninstalling an Optional SubPackage, page 4-16

Installing an Optional SubPackage

Note If you install the Cisco WebEx Node for ASR 1000 Series SPA before the software subpackage is installed, you will receive the error message %ASR1000_RP_SPA-3-MISSING_SPA_PKG_ERR. This message will resolve once you download and install the appropriate sipspawma subpackage and then reload the router in individual subpackage mode.

To run the router using an optional subpackage, perform the following steps for each RP in the system:

Step 1 Verify that the RP is running in individual subpackage mode and was booted from a provisioning file. For more information, see the "Managing and Configuring a Router to Run Using Individual SubPackages From a Consolidated Package" section on page 4-10.

Step 2 Verify that the version of the optional subpackage that you want to install is the same version as the software running on the active RP.

Step 3 Download the optional subpackage that you want to install. Optional subpackages must be downloaded independently from consolidated packages for the Cisco ASR 1000 Series Routers.

Step 4 On each RP, copy the optional subpackage to the directory where any other individual subpackages and the provisioning file is located.

Step 5 Run the **request platform software package install rp file** command, as shown in the following example.

Note Do not use the optional **slot** or **bay** keywords for the initial installation.

```
Router# request platform software package install rp 0 file
bootflash:asr1000rp1-sipspawmak9.BLD_V122_33_XND_ASR_RLS4_THROTTLE_LATEST_20090511_042308.
pkg
--- Starting local lock acquisition on R0 --- Finished local lock acquisition on R0

--- Starting file path checking ---
Finished file path checking

--- Starting image file verification --- Checking image file names Verifying image file
locations Locating image files and validating name syntax
   Found asr1000rp1-
sipspawmak9.BLD_V122_33_XND_ASR_RLS4_THROTTLE_LATEST_20090511_042308.pkg
Inspecting image file types
Processing image file constraints
Creating candidate provisioning file

WARNING: No package of type sipspawmak9 is installed.
WARNING: Package will be installed for all SIP slots and bays.

Finished image file verification

--- Starting candidate package set construction --- Verifying existing software set
Processing candidate provisioning file Constructing working set for candidate package set
Constructing working set for running package set Checking command output Constructing
merge of running and candidate packages Checking if resulting candidate package set would
be complete Finished candidate package set construction

--- Starting compatibility testing ---
Determining whether candidate package set is compatible
```

```
WARNING:
WARNING: Candidate software combination not found in compatibility database
WARNING:

Determining whether installation is valid

WARNING:
WARNING: Candidate software combination not found in compatibility database
WARNING:

Software sets are identified as compatible Checking IPC compatibility with running
software Checking candidate package set infrastructure compatibility Checking
infrastructure compatibility with running software Checking package specific compatibility
Finished compatibility testing

--- Starting impact testing ---
Checking operational impact of change
Finished impact testing

--- Starting list of software package changes --- No old package files removed New files
list:
   Added asr1000rp1-
sipspawmak9.BLD_V122_33_XND_ASR_RLS4_THROTTLE_LATEST_20090511_042308.pkg
Finished list of software package changes

--- Starting commit of software changes --- Updating provisioning rollback files Creating
pending provisioning file Committing provisioning file Finished commit of software changes

--- Starting analysis of software changes --- Finished analysis of software changes

--- Starting update running software --- Blocking peer synchronization of operating
information Creating the command set placeholder directory
   Finding latest command set
   Finding latest command shortlist lookup file
   Finding latest command shortlist file
   Assembling CLI output libraries
   Assembling CLI input libraries
   Applying interim IPC and database definitions
   Replacing running software
   Replacing CLI software
   Restarting software
   Restarting software: target frus filtered out ... skipped
   Applying final IPC and database definitions
   Generating software version information
   Notifying running software of updates
   Unblocking peer synchronization of operating information Unmounting old packages
Cleaning temporary installation files
   Finished update running software

SUCCESS: Finished installing software.
```

Uninstalling an Optional SubPackage

In Cisco IOS XE Release 2.4, the Cisco ASR 1000 Series Routers support removal of an installed optional subpackage.

To uninstall an optional subpackage, perform the following steps:

Step 1 On each RP in the system, use the **request platform software package uninstall** command as shown in the following example:

```
Router# request platform software package uninstall rp 0 type sipspawma
--- Starting local lock acquisition on R0 --- Finished local lock acquisition on R0

--- Starting uninstall preparation ---
Validating uninstall options
Creating candidate provisioning file
Finished uninstall preparation

--- Starting candidate package set construction --- Verifying existing software set
Processing candidate provisioning file Constructing working set for candidate package set
Constructing working set for running package set Checking command output Constructing
merge of running and candidate packages Checking if resulting candidate package set would
be complete Finished candidate package set construction

--- Starting compatibility testing ---
Determining whether candidate package set is compatible Determining whether installation
is valid Software sets are identified as compatible Checking IPC compatibility with
running software Checking candidate package set infrastructure compatibility Checking
infrastructure compatibility with running software Checking package specific compatibility
Finished compatibility testing

--- Starting list of software package changes --- Old files list:
   Removed asr1000rp1-
sipspawmak9.BLD_V122_33_XND_ASR_RLS4_THROTTLE_LATEST_20090511_042308.pkg
No new package files added
Finished list of software package changes

--- Starting commit of software changes --- Updating provisioning rollback files Creating
pending provisioning file Committing provisioning file Finished commit of software changes

--- Starting analysis of software changes --- Finished analysis of software changes

--- Starting update running software --- Blocking peer synchronization of operating
information Creating the command set placeholder directory
   Finding latest command set
   Finding latest command shortlist lookup file
   Finding latest command shortlist file
   Assembling CLI output libraries
   Assembling CLI input libraries
   Applying interim IPC and database definitions
   Replacing running software
   Replacing CLI software
   Restarting software
   Restarting software: target frus filtered out ... skipped
   Applying final IPC and database definitions
   Generating software version information
   Notifying running software of updates
   Unblocking peer synchronization of operating information Unmounting old packages
Cleaning temporary installation files
   Finished update running software

SUCCESS: Finished installing software.
```

Step 2 Use the **show version provisioned** command to verify that the optional subpackage is no longer installed.

Step 3 (Optional) Use the **request platform software clean** command to remove the unused packages.

Troubleshooting Software Mismatch with ESP Board ASR1000-ESP10-N

The Cisco ASR 1000 ESP board ASR1000-ESP10-N enables you to install a Cisco ASR 1000 Series Aggregation Services Router without encryption features. This unrestricted product format, in which all encryption chips have been removed, can be provided in the Cisco ASR 1002, 1004, and 1006 Series Routers. This ESP must be used with the appropriate Cisco IOS XE 2.2 image that does not contain encryption software.

You cannot load Cisco IOS XE software images or subpackages which contain encryption software, onto a Cisco ASR 1000 Series Router that contains the non crypto enabled ESP board (ASR1000-ESP10-N). As soon as the combination of an encryption-enabled Cisco IOS XE image and encryption-disabled ESP is detected—on bootup, online removal or insertion, or during an install—a message is emitted:

```
ESP[0|1] does not support strong cryptography. Chassis will reload.
```

The Route Processor (RP) then reloads the chassis and reboots continuously.

To troubleshoot this problem, follow these steps:

Step 1 Enable ROM Monitor (ROMmon) mode by entering the **reload** command.

Step 2 Press the Break key during the first 60 seconds while the system is booting.

Step 3 Force the system to remain in ROMmon mode, waiting for manual bootup, by entering the command: **confreg 0x0**.

Step 4 Load the latest Cisco IOS XE package or subpackages that are compatible with the ASR1000-ESP10-N ESP board.

Compatible subpackages include:

- Cisco ASR1000 Series RP1 IP BASE W/O CRYPTO
- Cisco ASR1000 Series RP1 ADVANCED IP SERVICES W/O CRYPTO
- Cisco ASR1000 Series RP1 ADVANCED ENTERPRISE SERVICES W/O CRYPTO

Step 5 Remove the forced manual boot mode by entering the command: **confreg** *config register setting*, where *config register setting* is the default for the user system, often 0x2102.

Step 6 Reboot the system.

Upgrading Individual SubPackages

One consolidated package contains seven individual subpackages. Each individual subpackage can be upgraded independently.

When using this method to upgrade subpackages, however, remember that all subpackages that are part of the same consolidated package should be used together on the Cisco ASR 1000 Series Routers. Individual subpackage upgrades are atypical on the Cisco ASR 1000 Series Routers, because it is very rare to experience a case where a single subpackage is upgraded without upgrading all the subpackages from the consolidated package. Individual subpackage upgrades are most useful when only a single subpackage of an otherwise functioning set of subpackages requires an upgrade.

This section uses the SPA subpackage installation to illustrate an individual subpackage upgrade. Although it does not comprehensively cover every possible individual subpackage upgrade scenario, the CLI for each type of upgrade is similar.

Upgrading a SPA SubPackage

In the following example, the **request platform software package install** command is used to upgrade a SIPSPA subpackage for the SPA in bay 0 of the SIP in router slot 1. In this example, the **force** option, which forces the upgrade past any prompt (such as already having the same subpackage installed), is used.

Caution Do not run your router using subpackages from different consolidated packages.
The Cisco ASR 1000 Series Router is not designed to use multiple subpackages from different consolidated packages. Running subpackages from different consolidated packages is not recommended and can cause unpredictable router behavior.
Individual subpackages should only be upgraded individually if you or a Cisco customer support representative has determined that upgrading to the same subpackage that is running could have some benefit for your Cisco ASR 1000 Series Router.

```
Router# request platform software package install rp 0 file
bootflash:asr1000rp1-sipspa.02.01.00.122-33.XNA.pkg slot 1 bay 0 force
--- Starting installation state synchronization ---
Finished installation state synchronization

--- Starting file path checking ---
Finished file path checking

--- Starting image file verification ---
Checking image file names
Verifying image file locations
Locating image files and validating name syntax
Inspecting image file types
Processing image file constraints
Creating candidate provisioning file
Finished image file verification

--- Starting candidate package set construction ---
Verifying existing software set
Processing candidate provisioning file

Constructing working set for candidate package set
Constructing working set for running package set
Checking command output
Constructing merge of running and candidate packages
Finished candidate package set construction

--- Starting compatibility testing ---
Determining whether candidate package set is compatible

WARNING:
WARNING: Candidate software combination not found in compatibility database
WARNING:

Determining whether installation is valid
Software sets are identified as compatible
Checking IPC compatibility with running software

Checking candidate package set infrastructure compatibility
Checking infrastructure compatibility with running software
Finished compatibility testing

--- Starting impact testing ---
Checking operational impact of change
```

```
Finished impact testing

--- Starting commit of software changes ---
Updating provisioning rollback files
Creating pending provisioning file

Committing provisioning file
Finished commit of software changes

--- Starting analysis of software changes ---
Finished analysis of software changes

--- Starting update running software ---
Blocking peer synchronization of operating information
Creating the command set placeholder directory
  Finding latest command set
  Assembling CLI output libraries
  Assembling CLI input libraries
  Applying interim IPC and database definitions
    interim IPC and database definitions applied
      Replacing running software
      Replacing CLI software
      Restarting software
Restarting SPA CC1/0
      Applying interim IPC and database definitions
      Notifying running software of updates
      Unblocking peer synchronization of operating information
    Unmounting old packages
    Cleaning temporary installation files
      Finished update running software

SUCCESS: Finished installing software.

Router#
```

Software Upgrade Processes Supported by Cisco ASR 1000 Series Routers

Cisco ASR 1000 Series Aggregation Services Routers support the following software upgrade procedures:

- In-Service Software Upgrades (ISSU) for redundant platforms—The ISSU process allows software to be updated or otherwise modified while packet forwarding continues with minimal interruption. ISSU supports two different software upgrade modes:
  - Consolidated package mode
  - Subpackage mode

  The supported platforms include Cisco ASR 1006 and Cisco ASR 1013 Routers.
- Upgrade process with service impact for non redundant platforms—Non-hardware-redundant chassis models (such as the Cisco ASR 1001 Router, Cisco ASR 1001-X Router, Cisco ASR 1002 Router, Cisco ASR 1002-X Router, and Cisco ASR 1004 Router) do not support ISSU upgrade or downgrade. Instead subpackage software upgrade is supported only if the router is running in subpackage mode.Traffic loss cannot be avoided during the installation of the ESP package as a part of ISSU.

Table 5-1 lists Compatibility matrix of the software upgrade process for various Cisco ASR 1000 Series Aggregation Services Router.

Table 5-1 Software Upgrade Compatibility Matrix for Various Cisco ASR 1000 Series Aggregation Services Router Platforms

Platform	Consolidated Package	SubPackage
Cisco ASR 1006 Router	Supported	Supported
CiscoASR 1013 Router	Supported	Supported
CiscoASR 1004 Router	Not Supported	Supported
CiscoASR 1002 Router	Not Supported	Supported
Cisco ASR 1002-X Router	Not Supported	Supported
CiscoASR 1002-F Router	Not Supported	Supported
CiscoASR 1001 Router	Not Supported	Supported
CiscoASR 1001-X Router	Not Supported	Supported

Finding Feature Information in This Module

Your Cisco IOS software release may not support all of the features documented in this module. To reach links to specific feature documentation in this module and to see a list of the releases in which each feature is supported, use the "Feature Information for Software Upgrade Process" section on page 5-251.

Finding Support Information for Platforms and Cisco IOS and Catalyst OS Software Images

Use Cisco Feature Navigator to find information about platform support and Cisco IOS and Catalyst OS software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Contents

This guide discusses various software upgrade procedures and contains the following sections:

- Prerequisites for Software Upgrade Processes, page 5-2
- ISSU Upgrade for Redundant Platforms, page 5-2
- Upgrade Process with Service Impact for Nonredundant Platforms, page 5-74
- Minimal Disruptive Restart ISSU, page 5-152

Prerequisites for Software Upgrade Processes

Be sure to complete the following prerequisites for running the ISSU process based on your chassis model:

- Refer to the ISSU compatibility tables in the *Release Notes for Cisco ASR 1000 Series Aggregation Services Routers.*
- 4 GB of DRAM memory is required for installing software upgrade on a system with RP1 route processor.
- ISSU is supported when the router is running in subpackage mode or in consolidated package mode.
- For the Cisco ASR 1001 Router, Cisco ASR 1001-X Router and Cisco ASR 1002-X Router, the Cisco IOS Software redundancy requires 8-GB DRAM and the IOS software redundancy license.

ISSU Upgrade for Redundant Platforms

ISSU represent a full or partial software upgrade of a system from one version to another with minimal outage on the forwarding plane (minimal packet loss) and no outage on the control plane.

This section covers the following topics:

- Overview of ISSU on the Cisco ASR 1000 Series Routers, page 5-3
- ISSU Upgrade Procedures, page 5-6
- In Service One-Shot Software Upgrade Procedure, page 5-71
- ISSU Procedures (Prior to Cisco IOS XE Release 2.1.2), page 5-72

Overview of ISSU on the Cisco ASR 1000 Series Routers

For the Cisco ASR 1000 Series Routers, ISSU-compatibility depends on the software subpackage being upgraded and the hardware configuration. Consolidated packages are ISSU-compatible in dual RP configurations only and have other limitations described later in this document. Some RP and ESP software subpackages can be upgraded in service even in single RP or ESP hardware configurations via dual IOS processes running on the RP; others require dual RP or ESP configurations for an ISSU upgrade. The SPA and SIP software subpackages must be upgraded on a per-SPA or per-SIP basis. See Table 5-2 to view an In Service table that addresses the contexts where limited interruption upgrades can be performed.

If you are updating multiple subpackages, you should also realize that the sequence of the upgrade is important to minimize router downtime for the software upgrade (see the "Using ISSU to Upgrade Subpackages (Prior to Cisco IOS XE Release 2.1.2)" section on page 5-72).

The specific procedures in this document represent supported and tested installation sequences. The Cisco IOS XE system software allows other installation sequences for special purposes under the guidance of Cisco customer support representatives, but the steps in this document should be followed otherwise. These steps should be followed completely, as the Cisco ASR 1000 Series Routers are designed to run one version of Cisco IOS XE for all consolidated packages and subpackages on an RP, and running subpackages from different versions of Cisco IOS XE can cause unexpected router behavior.

When performing ISSU upgrades on the Cisco ASR 1000 Series Routers, it is important to remember that minimal interruption upgrades can be performed using either the **issu** command set or the **request platform** command set, and that either command set can be used to perform limited interruption individual consolidated package or subpackage upgrades.

Note ROMmon images are downloaded separately from Cisco IOS XE images and have their own installation procedures, and are therefore not mentioned as part in this document as part of the ISSU upgrade procedure.

Table 5-2 provides a list of the Cisco ASR 1000 Series Routers subpackages and whether or not they can be upgraded without losing any network traffic in single and dual RP and ESP configurations using ISSU.

Table 5-2 Limited Interruption Upgrade Compatibility Table

Subpackage	Nonredundant RP and ESP	Redundant RP and ESP
Consolidated package (any)	No (Reload required)	Yes (RP switchover)
RPBase	No (RP Reload required)	Yes (RP switchover)
RPControl	Yes (in-service)	Yes (in-service)
RPAccess	Yes (in-service)	Yes (in-service)
RPIOS	Yes[1] (IOS software switchover)	Yes (RP switchover)
ESPBase	No[2] (ESP reload required)	Yes[3] (via ESP switchover)
ESPX86Base	No[4] (ESP reload required)	Yes[5] (via ESP switchover)
SIPBase[6]	No (SPAs in SIP do not forward traffic during upgrade)	Yes (With Minimal Disruptive Restart (MDR), traffic is forwarded with minimal interruption)

Table 5-2 Limited Interruption Upgrade Compatibility Table (continued)

Subpackage	Nonredundant RP and ESP	Redundant RP and ESP
SIPSPA[7]	No (SPAs in SIP do not forward traffic during upgrade)	Yes (With MDR, traffic is forwarded with minimal interruption)
ELCBase[8]	No (Built-in SPAs in an Ethernet Line Card (ELC) do not forward traffic during upgrade)	Yes (With MDR, traffic is forwarded with minimal interruption)
ELCSPA[9]	No (Built-in SPAs in an ELC do not forward traffic during upgrade)	Yes (With MDR, traffic is forwarded with minimal interruption)
NIM SSD[10]	No (NIM reload required)	NA
NGWIC T1E1[11]	No (T1E1 do not forward traffic during upgrade)	NA

1. Only supported if software redundancy is configured on the RP.
2. ESP has to reload to complete ESPBase subpackage upgrade. All router traffic will be lost during ESP upgrade.
3. An ESP switchover occurs as part of the upgrade, so traffic is forwarded with minimal interruption.
4. ESP has to reload to complete ESPBase subpackage upgrade. All router traffic will be lost during ESP upgrade and ESPX86Base is not included in the RP1 bundle.
5. An ESP switchover occurs as part of the upgrade, so traffic is forwarded with minimal interruption.ESPX86Base is not included in the RP1 bundle.
6. Any SIPBase upgrade will require the SPA interfaces to go down during the upgrade for all the SPAs in the SIP.
7. Any SIPSPA upgrade will require the SPA interfaces for that particular SPA to go down during the upgrade.
8. Any ELCBase upgrade will require the ELC interfaces to go down during the upgrade for all the SPAs in the SIP. ELCBase is not included in the RP1 bundle
9. Any ELCSPA upgrade will require the ELC interfaces for that particular SPA to go down during the upgrade. ELCSPA is not included in the RP1 bundle
10. NIM SSD is a built-in module available in ASR 1001-X platform. It is a slot for an extra harddisk, which helps increase the memory.
11. NGWIC T1E1 is a built-in module available in ASR 1001-X platform, used for serial interface. Any NGWIC T1E1 upgrade will require T1E1 interfaces for that particular SPA to go down during the upgrade.

ISSU Rollback Timer Overview

The Cisco ASR 1000 Series Router ISSU procedure has a rollback timer. Rollback timers are used for for ISSU procedures on all Cisco routers that support ISSU, but this section will provide a brief overview of ISSU rollback timers on the Cisco ASR 1000 Series Routers.

During ISSU, the rollback timer begins after the consolidated package or subpackage is loaded. If the upgrade does not move forward in the amount of time specified in the rollback timer, the configuration will automatically “roll back” to the previous configuration and the ISSU upgrade will be cancelled.

Upgrades using the **issu** command set and the **request platform** command set both have the rollback timer option. The **issu** command set always uses a rollback timer; the **request platform** command set does not use a rollback timer unless the **auto-rollback** option is used in the **request platform software package install** command line.

For the **issu** command set, the **issu acceptversion** command can be entered to stop the rollback timer without committing the upgrade during the ISSU upgrade. The **issu commitversion** command can be entered to stop the rollback timer and commit the ISSU upgrade.

For the **request platform** command set, the **request platform software package install rp** *slot* **commit** command must be entered to stop the rollback timer only in cases where the **auto-rollback** option is used.

The rollback timer for the **issu** command set can be configured by entering the **issu set rollback-timer** command. The rollback timer when used with the **request platform** command set is specified when you use the **auto-rollback** option when entering the **issu request platform software package install** command.

For ISSU upgrades on Cisco ASR 1000 Series Routers, it may be advisable to set long rollback times when the upgrade is being performed on routers with large configurations.

The amount of time left on the rollback timer during an ISSU upgrade can be checked by entering the **show issu rollback-timer** command.

Software Upgrade with Dual IOS Processes on a Single RP Overview

To complete a software upgrade of an individual subpackage using dual IOS processes on a single RP, SSO must first be enabled.

Software upgrade with dual IOS processes is useful for upgrading the individual RP subpackages that can be upgraded without a router reload. See Table 5-2 for a list of these subpackages. Importantly, note that most subpackage upgrades in a single RP configuration require a hardware reload to complete (whether an RP reload for an RP subpackage, an ESP reload for the ESPBase subpackage, a SIP reload for a SIPBase subpackage, or a SPA reload for the SIPSPA subpackage), so limited interruption upgrades for single RP configurations are not available in most upgrade scenarios.

For information on configuring a second IOS process on a single RP, see the "Using Subpackages for Software Upgrade on a Cisco ASR 1002 Router or Cisco ASR 1004 Router (software upgrade Command Set)" section on page 5-91.

Cisco IOS XE Software Package Compatibility for ISSU

When upgrading the Cisco IOS XE operating system software using the ISSU process, it is important to determine the compatibility of the upgraded software to your current software and hardware. The ISSU process allows software to be updated or otherwise modified while packet forwarding continues with minimal interruption.

Cisco IOS XE release compatibility using the ISSU process utilizes the SSO functionality to preserve state while software versions on the router differ, as during an upgrade. Most SSO-capable features in each Cisco IOS XE release are ISSU capable. ISSU is only supported if SSO is enabled in the configuration and the system is in a steady state (SSO ready state has been achieved). ISSU compatibility depends on the set of specific feature clients that are in use and whether they support ISSU. All ISSU upgrades include at least one IOS switchover operation. It is important to understand which features are in use and whether these features are ISSU compatible.

Cisco ASR1006 or ASR 1013 series routers are hardware-redundant chassis. The hardware-redundant chassis has two ESP linecards and two RPs which exchange state using hardware links. The Cisco ASR1002 and ASR1004 Series Routers are not hardware redundant, but are software-redundancy capable. The nonredundant chassis has a single RP and a single ESP, but allows the operation of up to two IOS processes on the single RP to exchange states locally.

Restrictions for ISSU

Restrictions for the ISSU procedures include:

Note Cisco IOS XE software compatibility is between "like" images, for example, *advipservicesk9* to *advipservicesk9*, *adventerprisek9* to *adventerprisek9*, and so on. Cross-image-type upgrades or installations are not supported in the ISSU process. For example, you cannot upgrade *ipbase* to *advipservicesk9* or *advipservices* to *advipservicesk9*.

- Different image types must not be run simultaneously.
- For ATM SPAs on the Cisco ASR1000 Series Routers, ISSU from releases prior to Cisco IOS XE Release 2.5.0 to Cisco IOS XE Release 2.5.0, or from Cisco IOS XE Release 2.5.0 to a release prior to Cisco IOS XE Release 2.5.0, is not supported. If you want to perform ISSU in this environment, you must first remove the configuration from the ATM SPAs on the router, and then shut down the SPAs using the **shutdown** command prior to running the ISSU process.
- Cisco IOS XE releases not listed as compatible in the ISSU compatibility tables (documented in the *Release Notes for Cisco ASR 1000 Series Aggregation Services Routers* as stated in the "Prerequisites for Software Upgrade Processes" section on page 5-2) must not be run simultaneously (in a Cisco ASR1006 series router or Cisco ASR 1013 series router) or co-installed on any of the Cisco ASR1000 Series Routers since unexpected failures of one or both RPs or state loss can be experienced. Cisco IOS XE releases listed as partially compatible may incur a loss of state. Cisco IOS XE releases listed as requiring an intermediate release are not directly compatible; however, a migration path is available to preserve some or all state by upgrading to a separate intermediate version, as shown in the tables. The tables do not cover nonredundant (software or hardware) environments as no incremental update is possible under those circumstances.
- In Cisco IOS XE Release 3.1S, ISSU upgrade and subpackage software upgrade from Cisco IOS XE Release 2.x.x to Cisco IOS XE Release 3.x.xS, including release 3.1S, are not supported. The ISSU downgrade from Cisco IOS XE Release 3.x.xS, including release 3.1S to 2.x.x, is also not supported. ISSU upgrade and subpackage software upgrade is restarted from Cisco IOS XE Release 3.1S. Therefore, rebuilds and releases after Cisco IOS XE Release 3.1S will support ISSU and software upgrade and downgrade, based on the ISSU compatibility matrix tables.

Note When you run the software upgrade from Cisco IOS XE Release 2.x.x to Cisco IOS XE Release 3.x.xS, you have to load the new image on both RPs, verify that it is good code, change the boot loader variable, and reboot the whole chassis. Failure to do that results in a "wedged" router and the only remedy is physically pull out one of the RPs, boot on the remaining RP, downgrade its code to the same version as the pulled out RP and start the process over again.

ISSU Upgrade Procedures

This section contains the following topics:

- Using ISSU to Perform a Consolidated Package Upgrade in a Dual Route Processor Configuration, page 5-7
- Using ISSU to Upgrade the Subpackages in a Dual Route Processor Configuration, page 5-13

Using ISSU to Perform a Consolidated Package Upgrade in a Dual Route Processor Configuration

Consolidated packages can only be upgraded using ISSU in dual Route Processor configurations. ISSU is not supported for consolidated package upgrades in single Route Processor configurations.

If you want the RPs on your Cisco ASR 1000 Series router to be running using a consolidated package after the ISSU upgrade is complete, use the following instructions:

Note This procedure will only work if the current RPs are already running consolidated packages.

SUMMARY STEPS

1. **ip tftp** *source-interface gigabitethernet port*
2. **copy tftp:** *URL-to-target-location*
3. **copy** *source-file-system***:***filename standby-destination-filesystem*
4. **dir** *URL-to-target-location*

   dir *URL-to-target-stby-location*
5. **issu loadversion rp** *upgrade-rp-number standby-file-system***:***filename*
6. **issu runversion**
7. **telnet** *ip-address port*
8. **issu acceptversion**
9. **issu commitversion**
10. **show version**, **show version** *active-RP* **running**, **show version** *active-RP* **provisioned**

    show platform

    show running-configuration
11. **hw-module slot** *RP slot number* **reload**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**ip tftp** *source-interface gigabitethernet slot/port* **Example:** `Router(config)# ip tftp source-interface gigabitethernet 0`	Specifies the Gigabit Ethernet TFTP source-interface to be configured: *slot/port*—Specifies the location of the TFTP source-interface. **Note** To copy a file using TFTP through the Management Ethernet interface, the **ip tftp source-interface GigabitEthernet 0** command must be entered before entering the **copy tftp** command.
Step 2	**copy tftp:** *URL-to-target-location* **copy** *source-file-system*:*filename standby-destination-filesystem* **Example:** `Router# copy tftp bootflash:`	Copy the consolidated package onto the active RP.
Step 3	**copy** *source-file-system*:*filename standby-destination-filesystem* **Example:** `Router# copy bootflash:asr1000rp1-adventerprisek9.02.01.01.122-33.XNA1.bin stby-bootflash:`	Copy the consolidated package onto the standby RP.

Step 4	**dir** *URL-to-target-location*
	dir *URL-to-target-stby-location*
	
	Example:
	`Router# dir bootflash:`
	`Router# dir stby-bootflash:`
		(Optional) Display the contents of the target directories to confirm the successful copy of the file package.

Step 5	**issu loadversion rp** *upgrade-rp-number standby-file-system*:*filename* **Example:** `Router# issu loadversion rp 1 file stby-bootflash:asr1000rp1-adventerprisek9.02.01.01.122-33.XNA1.bin`	Load the target consolidated package onto the standby RP. After you receive the message indicating that the terminal state has been reached, go on to Step 6.

	Command or Action	Purpose
Step 6	**issu runversion** **Example:** `Router# issu runversion`	Run the consolidated package that was loaded in Step 5. **Note** If this command is entered before the terminal state is reached, a "`peer is not online`" or "`Standby RP is not in terminal state`" error message will be seen and the **issu runversion** command will not work. If the **issu runversion** command does not run for these reasons, wait for the "`terminal state is reached`" message to appear and retry the **issu runversion** command. You can also monitor the terminal state using the **show platform** command. After ISSU runversion is completed, a switchover will automatically occur and the standby RP will become the active RP.

Step 7	**telnet** *ip-address port*
	
	Example:
	`[unix-server-1 ~]$ telnet 172.17.52.157 2003`
	`User Access Verification`
	
	`Username: user`
	`Password: ********`
	
	`Router>`
		Log in to the RP being upgraded, preferably using the RP's console port, to complete the upgrade. (This is the new active RP, that was the standby RP prior to the ISSU process.)
		
		Note Ensure the hostname does not end in "-stby" after logging into the RP, as this indicates that the RP being accessed is still the standby RP.
		
		There are many ways to log on to a console port. The example shows access to the console port from a UNIX host using telnet.

Step 8	**issu acceptversion** **Example:** `Router# issu acceptversion`	(Optional) Stops the ISSU rollback timer. This step is optional as long as Step 9 is completed before the rollback timer expires.
Step 9	**issu commitversion** **Example:** `Router# issu commitversion`	Completes the ISSU upgrade.

	Command or Action	Purpose
Step 10	**show version** **show version** *active-RP* **running** **show version** *active-RP* **provisioned** **show platform** **show running-configuration**	(Optional) Enter the **show version**, **show platform**, or **show running-configuration** commands to confirm that the upgrade completed successfully, as follows: • **show version**—Confirm that the correct software version is running on your router and that the RP was booted from the correct file. • **show running-configuration**—Confirm that the booting configuration, in particular the boot system statements, are correct. **Tip** Use the **show running-config | include boot** command to display only the boot statements from the **show running-configuration** output. • **show platform**— Confirm that both RP0 and RP1 are correctly running as active and standby.
Step 11	**hw-module slot** *RP-slot* **reload** **Example:** `Router# hw-module slot R0 reload`	Reload the new software on the Standby RP.

Examples

The following example shows how to perform consolidated package upgrade in a dual route processor configuration:

```
Router(config)# ip tftp source-interface gigabitethernet 0
Router# copy tftp bootflash:
Address or name of remote host []? 172.17.16.81
Source filename []?
/auto/tftp-users/user/asr1000rp1-adventerprisek9.02.01.01.122-33.XNA1.bin
Destination filename [asr1000rp1-adventerprisek9.02.01.01.122-33.XNA1.bin]?
Accessing
tftp://172.17.16.81//auto/tftp-users/user/asr1000rp1-adventerprisek9.02.01.01.122-33.XNA1.
bin...
Loading /auto/tftp-users/user/asr1000rp1-adventerprisek9.02.01.01.122-33.XNA1.bin from
172.17.16.81 (via GigabitEthernet0):
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!
[OK - 209227980 bytes]

209227980 bytes copied in 329.215 secs (635536 bytes/sec)

Router# copy bootflash:asr1000rp1-adventerprisek9.02.01.01.122-33.XNA1.bin stby-bootflash:
Destination filename [asr1000rp1-adventerprisek9.02.01.01.122-33.XNA1.bin]?
```

```
Copy in progress...CCCCCCCC
<output removed for brevity>
209227980 bytes copied in 434.790 secs (481216 bytes/sec)

Router# dir bootflash:
Directory of bootflash:/

   11  drwx       16384   Dec 4 2007 04:32:46 -08:00  lost+found
86401  drwx        4096   Dec 4 2007 06:06:24 -08:00  .ssh
14401  drwx        4096   Dec 4 2007 06:06:36 -08:00  .rollback_timer
28801  drwx        4096  Jul 21 2008 15:29:25 -07:00  .prst_sync
43201  drwx        4096   Dec 4 2007 04:34:45 -08:00  .installer
   12  -rw-   208904396  May 28 2008 16:17:34 -07:00
asr1000rp1-adventerprisek9.02.01.00.122-33.XNA.bin
57604  -rw-    47071436  May 29 2008 15:45:24 -07:00
asr1000rp1-espbase.02.01.00.122-33.XNA.pkg
57602  -rw-        5740  May 29 2008 15:45:23 -07:00
asr1000rp1-packages-adventerprisek9.02.01.00.122-33.XNA.conf
57605  -rw-    20334796  May 29 2008 15:45:25 -07:00
asr1000rp1-rpaccess.02.01.00.122-33.XNA.pkg
57606  -rw-    22294732  May 29 2008 15:45:25 -07:00
asr1000rp1-rpbase.02.01.00.122-33.XNA.pkg
57607  -rw-    21946572  May 29 2008 15:45:26 -07:00
asr1000rp1-rpcontrol.02.01.00.122-33.XNA.pkg
57608  -rw-    48099532  May 29 2008 15:45:27 -07:00
asr1000rp1-rpios-adventerprisek9.02.01.00.122-33.XNA.pkg
57609  -rw-    34324684  May 29 2008 15:45:28 -07:00
asr1000rp1-sipbase.02.01.00.122-33.XNA.pkg
57610  -rw-    22124748  May 29 2008 15:45:29 -07:00
asr1000rp1-sipspa.02.01.00.122-33.XNA.pkg
   14  -rw-      275093  May 29 2008 16:27:53 -07:00
crashinfo_RP_00_00_20080529-162753-DST
   15  -rw-        7516   Jul 2 2008 15:01:39 -07:00  startup-config
   13  -rw-       45977   Apr 9 2008 16:48:46 -07:00  target_support_output.tgz.tgz
   16  -rw-   209227980  Jul 17 2008 16:06:58 -07:00
asr1000rp1-adventerprisek9.02.01.01.122-33.XNA1.bin

928862208 bytes total (76644352 bytes free)

Router# dir stby-bootflash:
Directory of stby-bootflash:/

   11  drwx       16384  Dec 13 2004 03:45:47 -08:00  lost+found
87937  drwx        4096  Jul 17 2008 16:43:34 -07:00  .rollback_timer
14657  drwx        4096  Jul 17 2008 16:43:34 -07:00  .installer
29313  drwx        4096  Dec 13 2004 03:53:00 -08:00  .ssh
   12  -rw-    33554432  Dec 13 2004 03:53:49 -08:00  nvram_00100
   13  -rw-   208904396   Jun 5 2008 20:12:53 -07:00
asr1000rp1-adventerprisek9.02.01.00.122-33.XNA.bin
58625  drwx        4096  Jul 21 2008 15:32:59 -07:00  .prst_sync
43972  -rw-    47071436   Jun 5 2008 20:16:55 -07:00
asr1000rp1-espbase.02.01.00.122-33.XNA.pkg
43970  -rw-        5740   Jun 5 2008 20:16:54 -07:00
asr1000rp1-packages-adventerprisek9.02.01.00.122-33.XNA.conf
43973  -rw-    20334796   Jun 5 2008 20:16:56 -07:00
asr1000rp1-rpaccess.02.01.00.122-33.XNA.pkg
43974  -rw-    22294732   Jun 5 2008 20:16:56 -07:00
asr1000rp1-rpbase.02.01.00.122-33.XNA.pkg
43975  -rw-    21946572   Jun 5 2008 20:16:57 -07:00
asr1000rp1-rpcontrol.02.01.00.122-33.XNA.pkg
43976  -rw-    48099532   Jun 5 2008 20:16:58 -07:00
asr1000rp1-rpios-adventerprisek9.02.01.00.122-33.XNA.pkg
43977  -rw-    34324684   Jun 5 2008 20:16:59 -07:00
asr1000rp1-sipbase.02.01.00.122-33.XNA.pkg
```

```
43978  -rw-    22124748   Jun 5 2008 20:17:00 -07:00
asr1000rp1-sipspa.02.01.00.122-33.XNA.pkg
43971  -rw-        6256   Jun 5 2008 20:17:00 -07:00  packages.conf
   14  -rw-   209227980  Jul 17 2008 16:16:07 -07:00
asr1000rp1-adventerprisek9.02.01.01.122-33.XNA1.bin

945377280 bytes total (276652032 bytes free)

Router# issu loadversion rp 1 file
stby-bootflash:asr1000rp1-adventerprisek9.02.01.01.122-33.XNA1.bin
--- Starting installation state synchronization ---
Finished installation state synchronization

--- Starting file path checking ---
Finished file path checking

--- Starting system installation readiness checking ---
Finished system installation readiness checking

--- Starting installation changes ---
Setting up image to boot on next reset
Starting automatic rollback timer
Finished installation changes

SUCCESS: Software will now load.

PE23_ASR-1006#
*Jul 21 23:34:27.206: %ASR1000_OIR-6-OFFLINECARD: Card (rp) offline in slot R1
*Jul 21 23:34:27.271: %REDUNDANCY-3-STANDBY_LOST: Standby processor fault
(PEER_NOT_PRESENT)
*Jul 21 23:34:27.271: %REDUNDANCY-3-STANDBY_LOST: Standby processor fault (PEER_DOWN)
*Jul 21 23:34:27.271: %REDUNDANCY-3-STANDBY_LOST: Standby processor fault
(PEER_REDUNDANCY_STATE_CHANGE)
*Jul 21 23:37:05.528: %ASR1000_OIR-6-ONLINECARD: Card (rp) online in slot R1
*Jul 21 23:37:25.480: %REDUNDANCY-5-PEER_MONITOR_EVENT: Active detected a standby
insertion (raw-event=PEER_FOUND(4))

*Jul 21 23:37:25.480: %REDUNDANCY-5-PEER_MONITOR_EVENT: Active detected a standby
insertion (raw-event=PEER_REDUNDANCY_STATE_CHANGE(5))
Finished installation state synchronization
*Jul 21 23:37:26.349: %REDUNDANCY-3-IPC: IOS versions do not match.
*Jul 21 23:38:47.172: %HA_CONFIG_SYNC-6-BULK_CFGSYNC_SUCCEED: Bulk Sync succeeded
*Jul 21 23:38:47.173: %RF-5-RF_TERMINAL_STATE: Terminal state reached for (SSO)

Router# issu runversion
--- Starting installation state synchronization ---
Finished installation state synchronization

Initiating active RP failover
SUCCESS: Standby RP will now become active

PE23_ASR-1006#

System Bootstrap, Version 12.2(33r)XN2, RELEASE SOFTWARE (fc1)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 2008 by cisco Systems, Inc.

<additional output removed for brevity>

*Jul 21 23:43:31.970: %SYS-5-RESTART: System restarted --
Cisco IOS Software, IOS-XE Software (PPC_LINUX_IOSD-ADVENTERPRISEK9-M), Version
12.2(33)XNA, RELEASE SOFTWARE (fc1)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2008 by Cisco Systems, Inc.
```

```
Compiled Thu 01-May-08 00:29 by mcpre
*Jul 21 23:43:31.978: %SSH-5-ENABLED: SSH 1.99 has been enabled
*Jul 21 23:43:35.196: Relay: standby progression done
*Jul 21 23:43:35.197: %PLATFORM-6-RF_PROG_SUCCESS: RF state STANDBY HOT
```

At this point of the procedure, use your UNIX client to log in to the other RP:

```
[unix-server-1 ~]$ telnet 172.17.52.157 2003
User Access Verification

Username: user
Password: ********

Router>

Router# issu acceptversion
Cancelling rollback timer
SUCCESS: Rollback timer cancelled

Router# issu commitversion
--- Starting installation changes ---
Cancelling rollback timer
Saving image changes
Finished installation changes

Building configuration...
[OK]
SUCCESS: version committed: bootflash:asr1000rp1-adventerprisek9.02.01.01.122-33.XNA1.bin
```

After you confirm the software versions and configuration, reload the RP as shown in the following example:

```
Router# hw-module slot R0 reload
  Proceed with reload of module? [confirm]

  *Jul 21 23:54:27.271: %REDUNDANCY-3-STANDBY_LOST: Standby processor fault
(PEER_NOT_PRESENT)
  *Jul 21 23:54:27.271: %REDUNDANCY-3-STANDBY_LOST: Standby processor fault (PEER_DOWN)
  *Jul 21 23:54:27.271: %REDUNDANCY-3-STANDBY_LOST: Standby processor fault
(PEER_REDUNDANCY_STATE_CHANGE)

  *Jul 21 23:57:05.528: %ASR1000_OIR-6-ONLINECARD: Card (rp) online in slot R0
  *Jul 21 23:57:25.480: %REDUNDANCY-5-PEER_MONITOR_EVENT: Active detected a standby
insertion (raw-event=PEER_FOUND(4))

  *Jul 21 23:57:25.480: %REDUNDANCY-5-PEER_MONITOR_EVENT: Active detected a standby
insertion (raw-event=PEER_REDUNDANCY_STATE_CHANGE(5))

   *Jul 21 23:58:47.172: %HA_CONFIG_SYNC-6-BULK_CFGSYNC_SUCCEED: Bulk Sync succeeded
   *Jul 21 23:58:47.173: %RF-5-RF_TERMINAL_STATE: Terminal state reached for (SSO)
```

Using ISSU to Upgrade the Subpackages in a Dual Route Processor Configuration

This section provides instructions on performing an ISSU upgrade on a Cisco ASR 1000 Series Router with dual RPs that is currently running individual subpackages.

This section covers the following topics:

- Using ISSU to Upgrade the Subpackages on a Cisco ASR 1006 or ASR 1013 Router (issu Command Set), page 5-14
- Using ISSU to Upgrade Subpackages on a Cisco ASR 1006 Router or ASR 1013 Router (request platform command set), page 5-43

Using ISSU to Upgrade the Subpackages on a Cisco ASR 1006 or ASR 1013 Router (issu Command Set)

This section provides the instructions for performing an ISSU upgrade using subpackages on a Cisco ASR 1006 router or ASR 1013 router with a dual RP setup using the **issu** command set.

This procedure can only be performed if the current ASR 1006 or ASR 1013 routers have two active RPs and both RPs are running subpackages.

SUMMARY STEPS

1. **show version**

   show version *active-rp* **installed**

   dir *filesystem*:<*directory*>

   show platform

   show redundancy-states
2. **copy running-config startup-config**
3. **mkdir** *URL-to-directory-name*
4. **ip tftp** *source-interface gigabitethernet port*
5. **copy tftp:** *URL-to-target-location*
6. **request platform software package expand file** *URL-to-consolidated-package*
7. **dir** *URL-to-consolidated-package*
8. **copy** *file-system*:**asr1000rp2-espbase.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*

   copy *file-system*:**asr1000rp2-espx86base.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*

   copy *file-system*:**asr1000rp2-rpaccess.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*

   copy *file-system*:**asr1000rp2-rpbase.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*

   copy *file-system*:**asr1000rp2-rpcontrol.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*

   copy *file-system*:**asr1000rp2-rpios.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*

   copy *file-system*:**asr1000rp2-sipbase.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*

   copy *file-system*:**asr1000rp2-sipspa.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*

   copy *file-system*:**asr1000rp2-elcbase.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*

   copy *file-system*:**asr1000rp2-elcspa.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*

Note In step 8, each individual subpackage that was extracted in step 6 is copied to the directory where the subpackages that are currently running the active RP are stored.

9. **copy** *file-system*:**asr1000rp2-espbase.***version*.**pkg** *URL-to-directory-of-sub-packages-standby-RP*

   copy *file-system*:**asr1000rp2-esps86base.***version*.**pkg** *URL-to-directory-of-sub-packages-standby-RP*

   copy *file-system*:**asr1000rp2-rpaccess.***version*.**pkg** *URL-to-directory-of-sub-packages-standby-RP*

   copy *file-system*:**asr1000rp2-rpbase.***version*.**pkg** *URL-to-directory-of-sub-packages-standby-RP*

   copy *file-system*:**asr1000rp2-rpcontrol.***version*.**pkg** *URL-to-directory-of-sub-packages-standby-RP*

copy *file-system*:**asr1000rp2-rpios.***version*.**pkg** *URL-to-directory-of-sub-packages-standby-RP*

copy *file-system*:**asr1000rp2-sipbase.***version*.**pkg** *URL-to-directory-of-sub-packages-standby-RP*

copy *file-system*:**asr1000rp2-sipspa.***version*.**pkg** *URL-to-directory-of-sub-packages-standby-RP*

copy *file-system*:**asr1000rp2-elcbase.***version*.**pkg** *URL-to-directory-of-sub-packages-standby-RP*

copy *file-system*:**asr1000rp2-elcspa.***version*.**pkg** *URL-to-directory-of-sub-packages-standby-RP*

Note In step 9, each individual subpackage that was extracted in step 6 is copied to the directory where the subpackages that are currently running the standby RP are stored.

10. **issu loadversion rp** *standby-RP* **file** *URL-to-standby-file-system*:**asr1000rp****version**.**pkg force**
11. **hw-module slot** *standby-RP* **reload**
12. **issu loadversion rp** *active-RP* **file** *URL-to-active-file-system*:**asr1000rp2-{sipbase,sipspa}****version**.**pkg slot** *SIP-slot-number* **force**

    issu commitversion

    Repeat the step 12, for each available SIP installed in the router before moving onto the next step.
13. **issu loadversion rp** *active-RP* **file** *URL-to-active-file-system*:**asr1000rp2-{elcbase,elcspa}****version**.**pkg slot** *ELC-slot-number* **force**

    issu commitversion

    Repeat the step 13, for each available ELC installed in the router before moving onto the next step.
14. **issu loadversion rp** *active-RP* **file** *URL-to-active-file-system*:**asr1000rp2-esp****version**.**pkg slot** *standby-ESP-slot*

    issu commitversion

    issu loadversion rp *active-RP* **file** *URL-to-active-file-system*:**asr1000rp2-esp****version**.**pkg slot** *active-ESP-slot*

    issu commitversion
15. **issu loadversion rp** *active-RP* **file** *URL-to-active-file-system*:**asr1000rp2****version**.**pkg force**

    issu commitversion
16. **show version** *active-RP* **provisioned**

    show version *active-RP* **installed**
17. **redundancy force-switchover**
18. **request platform software package clean**

DETAILED STEPS

	Command or Action	Purpose

Step 1	**show version**
	show version *active-rp* **installed**
	dir *filesystem*: *directory*
	show platform
	show redundancy states
	
	Example:
	`Router# show version`
	`Router# show version r0 installed`
	`Router# dir bootflash:`
	`Router# show platform`
	`Router# show redundancy states`
		(Optional) Use the following commands to confirm the current router configuration, as follows:
		• **show version** and **show version** *active-rp* **installed**—Verify the running version of the Cisco IOS XE software on the router, and which file was used to boot the router, and where that file is stored.
		• **dir**—Confirm that the files that were used to boot the router are located in the directory.
		• **show platform**—Confirm the current status of the active and standby RPs.
		• **show redundancy states**—Confirm the operational and configured redundancy states.

Step 2	**copy running-config startup-config** **Example:** `Router# copy running-config startup-config`	After you have confirmed that the system states are acceptable, save the current configuration to the startup configuration.
Step 3	**mkdir** *URL-to-directory-name* **Example:** `Router# mkdir usb0:221subs`	Create a directory to store the consolidated package and subpackages. This directory must be created in most cases because the consolidated packages and subpackages have to be separated from the subpackages that booted the router at this step of the procedure.
Step 4	**ip tftp** *source-interface gigabitethernet port* **Example:** `Router(config)# ip tftp source-interface gigabitethernet 0`	Specifies the Gigabit Ethernet TFTP source-interface to be configured: *slot/port*—Specifies the location of the TFTP source-interface. **Note** To copy a file using TFTP through the Management Ethernet interface, the **ip tftp source-interface GigabitEthernet 0** command must be entered before entering the **copy tftp** command.
Step 5	**copy tftp:** *URL-to-target-location* **Example:** `Router# copy tftp: usb0:221subs`	Copy the consolidated package file into the directory created in Step 3. The consolidated package in this step should not be copied into the same directory where the subpackages that are currently running your router are stored (the directory containing the packages.conf provisioning file from which the router was booted). **Tip** It is recommended that you copy the package onto a usb: or harddisk: file system for space considerations when performing this step of the procedure.

	Command or Action	Purpose
Step 6	**request platform software package expand file** *URL-to-consolidated-package* **Example:** `Router# request platform software package expand file usb0:221subs/asr1000rp2-adventerprisek9.03.13.00.S.154-3.S-ext.bin`	Extract the subpackages out of the consolidated package file into the temporary directory. **Note** Take extra care to extract the subpackages to a temporary subdirectory and do not delete any of the files currently running the router at this point of the procedure. To erase the files that were running on the router before the ISSU upgrade, enter the **request platform software package clean** command after the ISSU upgrade has been completed.
Step 7	**dir** *target-URL* **Example:** `Router# dir usb0:221subs`	(Optional) Display the directory to confirm that the files were extracted.

Command or Action	Purpose

Step 8 **copy** *file-system*:**asr1000rp2-espbase.**_version_.**pkg** *URL-to-directory-of-sub-packages-active-RP*
copy *file-system*:**asr1000rp2-espx86base.**_version_.**pkg** *URL-to-directory-of-sub-packages-active-RP*
copy *file-system*:**asr1000rp2-rpaccess.**_version_.**pkg** *URL-to-directory-of-sub-packages-active-RP*
copy *file-system*:**asr1000rp2-rpbase.**_version_.**pkg** *URL-to-directory-of-sub-packages-active-RP*
copy *file-system*:**asr1000rp2-rpcontrol.**_version_.**pkg** *URL-to-directory-of-sub-packages-active-RP*
copy *file-system*:**asr1000rp2-rpios.**_version_.**pkg** *URL-to-directory-of-sub-packages-active-RP*
copy *file-system*:**asr1000rp2-sipbase.**_version_.**pkg** *URL-to-directory-of-sub-packages-active-RP*
copy *file-system*:**asr1000rp2-sipspa.**_version_.**pkg** *URL-to-directory-of-sub-packages-active-RP*
copy *file-system*:**asr1000rp2-elcbase.**_version_.**pkg** *URL-to-directory-of-sub-packages-active-RP*
copy *file-system*:**asr1000rp2-elcspa.**_version_.**pkg** *URL-to-directory-of-sub-packages-active-RP*
 	
Example:
`Router# copy usb0:221subs/asr1000rp2-espbase.03.13.00.S.154-3.S-ext.pkg bootflash:`
 	
`usb0:221subs/asr1000rp2-espx86base.03.13.00.S.154-3.S-ext.pkg bootflash:`
 	
`Router# copy usb0:221subs/asr1000rp2-rpaccess.03.13.00.S.154-3.S-ext.pkg bootflash:`
 	
`Router# copy usb0:221subs/asr1000rp2-rpbase.03.13.00.S.154-3.S-ext.pkg bootflash:`
 	
`Router# copy usb0:221subs/asr1000rp2-rpcontrol.03.13.00.S.154-3.S-ext.pkg bootflash:`
 	
`Router# copy usb0:221subs/asr1000rp2-rpios-adventerprisek9.03.13.00.S.154-3.S-ext.pkg bootflash:`
 	
`Router# copy usb0:221subs/asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg bootflash:`
 	
`Router# copy usb0:221subs/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg bootflash:`
 	
`Router# copy usb0:221subs/asr1000rp2-elcbase.03.13.00.S.154-3.S-ext.pkg bootflash:`
 	
`Router# copy usb0:221subs/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg bootflash:`
	Copy the subpackages out of the temporary directory into the directory on the router where the subpackages running the active RP are currently stored.
	
	Note Make sure to copy all the subpackage files extracted in step 6 to the current active packages directory.

Command or Action	Purpose

Step 9 **copy** *file-system*:**asr1000rp2-espbase.**_version_.**pkg** *URL-to-directory-of-sub-packages-standby-RP*
copy *file-system*:**asr1000rp2-espx86base.**_version_.**pkg** *URL-to-directory-of-sub-packages-standby-RP*
copy *file-system*:**asr1000rp2-rpaccess.**_version_.**pkg** *URL-to-directory-of-sub-packages-standby-RP*
copy *file-system*:**asr1000rp2-rpbase.**_version_.**pkg** *URL-to-directory-of-sub-packages-standby-RP*
copy *file-system*:**asr1000rp2-rpcontrol.**_version_.**pkg** *URL-to-directory-of-sub-packages-standby-RP*
copy *file-system*:**asr1000rp2-rpios.**_version_.**pkg** *URL-to-directory-of-sub-packages-standby-RP*
copy *file-system*:**asr1000rp2-sipbase.**_version_.**pkg** *URL-to-directory-of-sub-packages-standby-RP*
copy *file-system*:**asr1000rp2-sipspa.**_version_.**pkg** *URL-to-directory-of-sub-packages-standby-RP*
copy *file-system*:**asr1000rp2-elcbase.**_version_.**pkg** *URL-to-directory-of-sub-packages-standby-RP*
copy *file-system*:**asr1000rp2-elcspa.**_version_.**pkg** *URL-to-directory-of-sub-packages-standby-RP*
 	
Example:
`Router# copy usb0:221subs/asr1000rp2-espbase.03.13.00.S.154-3.S-ext.pkg stby-bootflash:`
 	
`Router# copy usb0:221subs/asr1000rp2-espx86base.03.13.00.S.154-3.S-ext.pkg stby-bootflash:`
 	
`Router# copy usb0:221subs/asr1000rp2-rpaccess.03.13.00.S.154-3.S-ext.pkg stby-bootflash:`
 	
`Router# copy usb0:221subs/asr1000rp2-rpbase.03.13.00.S.154-3.S-ext.pkg stby-bootflash:`
 	
`Router# copy usb0:221subs/asr1000rp2-rpcontrol.03.13.00.S.154-3.S-ext.pkg stby-bootflash:`
 	
`Router# copy usb0:221subs/asr1000rp2-rpios-adventerprisek9.03.13.00.S.154-3.S-ext.pkg stby-bootflash:`
 	
`Router# copy usb0:221subs/asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg stby-bootflash:`
 	
`Router# copy usb0:221subs/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg stby-bootflash:`
 	
`Router# copy usb0:221subs/asr1000rp2-elcbase.03.13.00.S.154-3.S-ext.pkg stby-bootflash:`
 	
`Router# copy usb0:221subs/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg stby-bootflash:`
	Copy the subpackages out of the temporary directory into the directory on the router where the subpackages running the standby RP are currently stored.
	
	Note Make sure to copy all the subpackage files extracted in step 6 to the current active packages directory.

	Command or Action	Purpose
Step 10	**issu loadversion rp** *standby-RP* **file** *target-standbyRP-URL-for-sub-packages***:asr1000rp*** *version****.pkg force** **Example:** `Router# issu loadversion rp 1 file stby-bootflash:asr1000rp*03.13.00.S.154-3.S-ext *.pkg force`	Upgrade the RP subpackages on the standby RP, where the "**rp***" wildcard is specified to capture all of the RP subpackages for the desired upgrade release.
Step 11	**hw-module slot** *standby-RP* **reload** **Example:** `Router# hw-module slot R1 reload`	Reload the standby RP.
Step 12	**issu loadversion rp** *active-RP* **file** *URL-to-active-file-system***:asr1000rp2-{sipbase,sipspa}*** *version****.pkg slot** *SIP-slot-number* **force** **issu commitversion** **Example:** `Router# issu loadversion rp 0 file bootflash:asr1000rp2-{sipbase,sipspa}*03.13.00.S.154-3.S-ext*.pkg slot 0 force`	Upgrade the SIP and SPA subpackages for each SIP on the router. **Note** This step must be completed one SIP at a time, and repeated for each SIP installed on the router before performing the next step. **Tip** You can use the **show ip interface brief** command to identify which slots contain SIPs and SPAs. The interfaces with three numbers (in the form *SIP-number*/*SPA-number*/*interface-number*) identify the SIP and SPA locations in the router. **Note** The *pattern* options used in this CLI (**sipbase** and **sipspa**) were introduced in Cisco IOS XE Release 2.1.2 and are not available in previous Cisco IOS XE Releases. See the "ISSU Procedures (Prior to Cisco IOS XE Release 2.1.2)" section on page 5-72 for pre-Cisco IOS XE Release 2.1.2 ISSU upgrade procedures.
Step 13	**issu loadversion rp** *active-RP* **file** *URL-to-active-file-system***:asr1000rp2-{elcbase,elcspa}*** *version****.pkg slot** *SIP-slot-number* **force** **issu commitversion** Repeat this step for each ELC installed in the router before moving onto the next step. **Example:** `Router# issu loadversion rp 0 file bootflash:asr1000rp2-{elcbase,elcspa}*03.13.00.S.154-3.S-ext*.pkg slot 0 force`	Upgrade the ELC and SPA subpackages for each ELC on the router. **Note** This step must be completed for one ELC at a time, and repeated for each ELC installed on the router before performing the next step. **Tip** You can use the **show ip interface brief** command to identify which slots contain ELCs and SPAs. The interfaces with three numbers (in the form *ELC-number*/*SPA-number*/*interface-number*) identify the ELC and SPA locations in the router. **Note** The *pattern* options used in this CLI (**elcbase** and **elcspa**) were introduced in Cisco IOS XE Release 3.10S and are not available in previous Cisco IOS XE Releases.

	Command or Action	Purpose

Step 14	**issu loadversion rp** *active-RP* **file** *URL-to-active-file-system*:**asr1000rp2-esp***version****.pkg slot** *standby*-ESP-*slot*
	issu commitversion
	issu loadversion rp *active-RP* **file** *URL-to-active-file-system*:**asr1000rp2-esp***version****.pkg slot** *active*-ESP-*slot*
	issu commitversion
	
	Example:
	`Router# issu loadversion rp 0 file bootflash:asr1000rp2-esp*03.13.00.S.154-3.S-ext*.pkg slot 1`
	`Router# issu commitversion`
	`Router# issu loadversion rp 0 file bootflash:asr1000rp2-esp*03.13.00.S.154-3.S-ext*.pkg slot 0`
	`Router# issu commitversion`
		Upgrade the ESP Base subpackage on the standby and the active ESPs.
		
		After entering the **issu loadversion rp** command on the active RP, the ESP switchover will occur automatically. Minimal traffic interruption will occur as a result of this switchover.

Step 15	**issu loadversion rp** *active-RP* **file** *URL-to-active-file-system*:**asr1000rp***version****.pkg force**
	issu commitversion
	
	Example:
	`Router# issu loadversion rp 0 file bootflash:asr1000rp2*03.13.00.S.154-3.S-ext*.pkg force`
	`issu commitversion`
		Upgrade all of the subpackages on the active RP.
		
		Note This step is required to ensure that all subpackages on the router were upgraded as part of this procedure, and might upgrade some subpackages that would otherwise be missed in the process.

Step 16	**show version** *active-RP* **provisioned**
	show version *active-RP* **installed**
	
	Example:
	`Router# show version r0 provisioned`
	`Router# show version r0 installed`
		(Optional) Confirm that the subpackages are provisioned and installed.

Step 17	**redundancy force-switchover** **Example:** `Router# redundancy force-switchover`	Force an RP switchover to complete the upgrade.
Step 18	**request platform software package clean** **Example:** `Router# request platform software package clean`	(Optional) Removes all unused subpackages files from the router.

Examples

The following example shows ISSU upgrade using subpackages on a Cisco ASR 1006 router or ASR 1013 router with a dual RP setup using the **issu** command set

```
Router# show version
Cisco IOS Software, IOS-XE Software (X86_64_LINUX_IOSD-ADVENTERPRISEK9-M), Version
15.3(2)S, RELEASE SOFTWARE (fc1)

<output removed for brevity>

System image file is "bootflash:Active_Dir/packages.conf"
```

```
<output removed for brevity>

cisco ASR1013 (RP2) processor with 4208889K/6147K bytes of memory.
Processor board ID FOX1343GJGC 20 Gigabit Ethernet interfaces
6 Ten Gigabit Ethernet interfaces
32768K bytes of non-volatile configuration memory.
8388608K bytes of physical memory.
1925119K bytes of eUSB flash at bootflash:.
78085207K bytes of SATA hard disk at harddisk:.

Configuration register is 0x2102

Router# show platform
Chassis type: ASR1013

Slot      Type                State                 Insert time (ago)
--------- ------------------- --------------------- -----------------
 2        ASR1000-SIP40       ok                    1d03h
 2/0      SPA-1X10GE-L-V2     ok                    1d03h
 2/1      SPA-1X10GE-L-V2     ok                    1d03h
 2/2      SPA-1X10GE-L-V2     ok                    1d03h
 2/3      SPA-1X10GE-L-V2     ok                    1d03h
 4        ASR1000-2T+20X1GE   ok                    1d03h
 4/0      BUILT-IN-2T+20X1GE  ok                    1d03h
 R0       ASR1000-RP2         ok, active            1d03h
 R1       ASR1000-RP2         ok, standby           1d03h
 F0       ASR1000-ESP100      ok, active            1d03h
 F1       ASR1000-ESP100      ok, standby           1d03h
 P0       ASR1013-PWR-AC      ok                    1d03h
 P1       ASR1013-PWR-AC      ok                    1d03h
 P2       ASR1013-PWR-AC      ok                    1d03h
 P3       ASR1013-PWR-AC      ps, fail              1d03h

Slot      CPLD Version        Firmware Version
--------- ------------------- ----------------------------------------
 2         00200800            15.3(3r)S
 4         00200800            15.2(1r)S
 R0        10021901            15.3(3r)S
 R1        10021901            15.3(3r)S
 F0        12071700            15.3(3r)S
 F1        12071700            15.3(3r)S

Router# show version r0 installed
Package: Provisioning File, version: n/a, status: active
  File: bootflash:Active_Dir/packages.conf, on: RP0
  Built: n/a, by: n/a
  File SHA1 checksum: a624f70f68c60292f4482433f43afd92487a55c4

Package: rpbase, version: 03.12.01.S.154-2.S, status: active
  File: bootflash:Active_Dir/asr1000rp2-rpbase.03.12.01.S.154-2.S.pkg, on: RP0
  Built: 2013-03-25_18.48, by: mcpre
  File SHA1 checksum: 3a9675142898cfac350d4e42f0e37bd9f4e48538

Package: rpcontrol, version: 03.12.01.S.154-2.S, status: active
  File: bootflash:Active_Dir/asr1000rp2-rpcontrol.03.12.01.S.154-2.S.pkg, on: RP0/0
  Built: 2013-03-25_18.48, by: mcpre
  File SHA1 checksum: 87b11f863f67fdf2610ee0769b929baab4c3efad

<output removed for brevity>

Router# dir bootflash:Active_Dir
Directory of bootflash:/Active_Dir/
```

```
   20  -rw-    41104112   Aug 3 2013 15:05:40 +05:30
asr1000rp2-elcbase.03.12.01.S.154-2.S.pkg
   21  -rw-    50285296   Aug 3 2013 15:05:40 +05:30
asr1000rp2-elcspa.03.12.01.S.154-2.S.pkg
   22  -rw-    82514676   Aug 3 2013 15:05:40 +05:30
asr1000rp2-espbase.03.12.01.S.154-2.S.pkg
   23  -rw-   101084628   Aug 3 2013 15:05:40 +05:30
asr1000rp2-espx86base.03.12.01.S.154-2.S.pkg
   17  -rw-        9059   Aug 3 2013 15:05:40 +05:30
asr1000rp2-packages-adventerprisek9.03.12.01.S.154-2.S.conf
   24  -rw-    29012724   Aug 3 2013 15:05:40 +05:30
asr1000rp2-rpaccess.03.12.01.S.154-2.S.pkg
   25  -rw-    49898964   Aug 3 2013 15:05:40 +05:30
asr1000rp2-rpbase.03.12.01.S.154-2.S.pkg
   26  -rw-    46557940   Aug 3 2013 15:05:40 +05:30
asr1000rp2-rpcontrol.03.12.01.S.154-2.S.pkg
   27  -rw-   114612988   Aug 3 2013 15:05:41 +05:30
asr1000rp2-rpios-adventerprisek9.03.12.01.S.154-2.S.pkg
   28  -rw-    41954036   Aug 3 2013 15:05:41 +05:30
asr1000rp2-sipbase.03.12.01.S.154-2.S.pkg
   29  -rw-    60957428   Aug 3 2013 15:05:41 +05:30
asr1000rp2-sipspa.03.12.01.S.154-2.S.pkg
   19  -rw-        9838   Aug 3 2013 15:05:41 +05:30
packages.conf

1940303872 bytes total (503164928 bytes free)


Router# show redundancy states
       my state = 13 -ACTIVE
       peer state = 8  -STANDBY HOT
       Mode = Duplex
       Unit = Primary
       Unit ID = 48

Redundancy Mode (Operational) = sso
Redundancy Mode (Configured)  = sso
Redundancy State              = sso
Maintenance Mode = Disabled
Manual Swact = enabled
Communications = Up
client count = 108
client_notification_TMR = 30000 milliseconds
RF debug mask = 0x0

Router# copy running-config startup-config
Destination filename [startup-config]?
Building configuration...
[OK]

Router# mkdir harddisk:Target_Subs
Create directory filename [Target_Subs]?
Created dir harddisk:/Target_Subs



Router# request platform software package expand file
harddisk:Target_Subs/asr1000rp2-adventerprisek9.03.13.00.S.154-3.S-ext.bin to
harddisk:Target_Subs
Verifying parameters
Validating package type
Copying package files
SUCCESS: Finished expanding all-in-one software package.
```

```
Router# dir harddisk:Target_Subs
Directory of harddisk:/Target_Subs/

3358722  -rw-   569597380   Aug 4 2013 18:45:38 +05:30
asr1000rp2-adventerprisek9.03.13.00.S.154-3.S-ext.bin
7684099  -rw-    37557200   Aug 4 2013 18:46:43 +05:30
asr1000rp2-elcbase.03.13.00.S.154-3.S-ext.pkg
7684100  -rw-    51194832   Aug 4 2013 18:46:43 +05:30
asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg
7684101  -rw-    80657364   Aug 4 2013 18:46:43 +05:30
asr1000rp2-espbase.03.13.00.S.154-3.S-ext.pkg
7684102  -rw-    95446456   Aug 4 2013 18:46:43 +05:30
asr1000rp2-espx86base.03.13.00.S.154-3.S-ext.pkg
7684097  -rw-        9381   Aug 4 2013 18:46:43 +05:30
asr1000rp2-packages-adventerprisek9.03.13.00.S.154-3.S-ext.conf
7684103  -rw-    23350232   Aug 4 2013 18:46:43 +05:30
asr1000rp2-rpaccess.03.13.00.S.154-3.S-ext.pkg
7684104  -rw-    37694900   Aug 4 2013 18:46:44 +05:30
asr1000rp2-rpbase.03.13.00.S.154-3.S-ext.pkg
7684105  -rw-    45536216   Aug 4 2013 18:46:44 +05:30
asr1000rp2-rpcontrol.03.13.00.S.154-3.S-ext.pkg
7684106  -rw-   118754284   Aug 4 2013 18:46:44 +05:30
asr1000rp2-rpios-adventerprisek9.03.13.00.S.154-3.S-ext.pkg
7684107  -rw-    38380500   Aug 4 2013 18:46:44 +05:30
asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg
7684108  -rw-    61760468   Aug 4 2013 18:46:44 +05:30
asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg
7684098  -rw-       10165   Aug 4 2013 18:46:44 +05:30  packages.conf

78704144384 bytes total (9254879232 bytes free)


Router# copy harddisk:Target_Subs/asr1000rp2-espbase.03.13.00.S.154-3.S-ext.pkg bootflash:
Active_Dir/Destination filename [Active_Dir/asr1000rp2-
espbase.03.13.00.S.154-3.S-ext.pkg]?
Copy in

progress...CCCC
```

```
progress...CCCC
```

```
Copy in
progress...CCCC
```

```
Router# copy harddisk:Target_Subs/asr1000rp2-rpcontrol.03.13.00.S.154-3.S-ext.pkg
stby-bootflash:
Destination filename [Active_Dir/asr1000rp2-rpcontrol.03.13.00.S.154-3.S-ext.pkg]?
Copy in
progress...CCCC
```

```
progress...CCCC
```

```
Checking infrastructure compatibility with running software
Checking infrastructure compatibility with running software ... skipped
Checking package specific compatibility
Finished compatibility testing
SUCCESS: Software is ISSU compatible.

Router#
Router#
Router# issu loadversion rp 1 file
stby-bootflash:Active_Dir/asr1000rp*03.13.00.S.154-3.S-ext*.pkg force
--- Starting local lock acquisition on R0 ---
Finished local lock acquisition on R0

--- Starting installation state synchronization ---
Finished installation state synchronization

--- Starting local lock acquisition on R1 ---
Finished local lock acquisition on R1

--- Starting file path checking ---
Finished file path checking

--- Starting image file verification ---
Checking image file names
Locating image files and validating name syntax
  Found asr1000rp2-elcbase.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-espbase.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-espx86base.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-rpaccess.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-rpbase.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-rpcontrol.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-rpios-adventerprisek9.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg
Verifying image file locations
Inspecting image file types
    WARNING: In-service installation of IOSD package
    WARNING: requires software redundancy on target RP
    WARNING: or on-reboot parameter
    WARNING: Automatically setting the on-reboot flag
    WARNING: In-service installation of RP Base package
    WARNING: requires software reboot of target RP
Processing image file constraints
Creating candidate provisioning file
Finished image file verification

--- Starting candidate package set construction ---
Verifying existing software set
Processing candidate provisioning file
Constructing working set for candidate package set
Constructing working set for running package set
Checking command output
Constructing merge of running and candidate packages
Checking if resulting candidate package set would be complete
Finished candidate package set construction

--- Starting compatibility testing ---
Determining whether candidate package set is compatible
Determining whether installation is valid
Determining whether installation is valid ... skipped
Verifying image type compatibility
Checking IPC compatibility for candidate software
Checking candidate package set infrastructure compatibility
```

```
Checking infrastructure compatibility with running software
Checking infrastructure compatibility with running software ... skipped
Checking package specific compatibility
Finished compatibility testing

--- Starting list of software package changes ---
Old files list:
  Removed asr1000rp2-elcbase.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-elcspa.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-espbase.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-espx86base.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-rpaccess.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-rpbase.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-rpcontrol.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-rpios-adventerprisek9.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-sipbase.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-sipspa.03.12.01.S.154-2.S.pkg
New files list:
  Added asr1000rp2-elcbase.03.13.00.S.154-3.S-ext.pkg
  Added asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg
  Added asr1000rp2-espbase.03.13.00.S.154-3.S-ext.pkg
  Added asr1000rp2-espx86base.03.13.00.S.154-3.S-ext.pkg
  Added asr1000rp2-rpaccess.03.13.00.S.154-3.S-ext.pkg
  Added asr1000rp2-rpbase.03.13.00.S.154-3.S-ext.pkg
  Added asr1000rp2-rpcontrol.03.13.00.S.154-3.S-ext.pkg
  Added asr1000rp2-rpios-adventerprisek9.03.13.00.S.154-3.S-ext.pkg
  Added asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg
  Added asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg
Finished list of software package changes

--- Starting commit of software changes ---
Updating provisioning rollback files
Creating pending provisioning file
Committing provisioning file
Finished commit of software changes

SUCCESS: Software provisioned.  New software will load on reboot.
Router#
Router# hw-module slot r1 reload
Proceed with reload of module? [confirm]
Router#
*Aug  4 19:14:01.721 IST: %IOSXE_OIR-6-OFFLINECARD: Card (rp) offline in slot R1
*Aug  4 19:14:01.761 IST: %REDUNDANCY-3-STANDBY_LOST: Standby processor fault
(PEER_NOT_PRESENT)
*Aug  4 19:14:01.761 IST: %REDUNDANCY-3-STANDBY_LOST: Standby processor fault (PEER_DOWN)
*Aug  4 19:14:01.761 IST: %REDUNDANCY-3-STANDBY_LOST: Standby processor fault
(PEER_REDUNDANCY_STATE_CHANGE)
*Aug  4 19:14:03.584 IST: %RF-5-RF_RELOAD: Peer reload. Reason: EHSA standby down
*Aug  4 19:14:03.594 IST: % Redundancy mode change to SSO

Router#
*Aug  4 19:17:35.443 IST: %IOSXE_OIR-6-ONLINECARD: Card (rp) online in slot R1
Router#
*Aug  4 19:17:48.061 IST: %REDUNDANCY-5-PEER_MONITOR_EVENT: Active detected a standby
insertion (raw-event=PEER_FOUND(4))

*Aug  4 19:17:48.061 IST: %REDUNDANCY-5-PEER_MONITOR_EVENT: Active detected a standby
insertion (raw-event=PEER_REDUNDANCY_STATE_CHANGE(5))

*Aug  4 19:17:51.003 IST: %REDUNDANCY-3-IPC: IOS versions do not match.

*Aug  4 19:19:08.380 IST: %NBAR_HA-5-NBAR_INFO: NBAR sync DONE!
*Aug  4 19:19:08.797 IST: %HA_CONFIG_SYNC-6-BULK_CFGSYNC_SUCCEED: Bulk Sync succeeded
*Aug  4 19:19:08.798 IST: %RF-5-RF_TERMINAL_STATE: Terminal state reached for (SSO)
```

```
Router# issu loadversion rp 0 file
bootflash:Active_Dir/asr1000rp2-{sipbase,sipspa}*03.13.00.S.154-3.S-ext*.pkg slot 2 force
--- Starting local lock acquisition on R0 ---
Finished local lock acquisition on R0

--- Starting installation state synchronization ---
Finished installation state synchronization

--- Starting file path checking ---
Finished file path checking

--- Starting image file verification ---
Checking image file names
Locating image files and validating name syntax
  Found asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg
Verifying image file locations
Inspecting image file types
Processing image file constraints
Creating candidate provisioning file
Finished image file verification

--- Starting candidate package set construction ---
Verifying existing software set
Processing candidate provisioning file
Constructing working set for candidate package set
Constructing working set for running package set
Checking command output
Constructing merge of running and candidate packages
Checking if resulting candidate package set would be complete
Finished candidate package set construction

--- Starting compatibility testing ---
Determining whether candidate package set is compatible

WARNING:
WARNING: Candidate software combination not found in compatibility database
WARNING:

Determining whether installation is valid
Creating matrix_file by locate_latest_matrix_file  /tmp/issu/provision/sw

WARNING:
WARNING: Candidate software combination not found in compatibility database
WARNING:


WARNING:
WARNING: Candidate software combination not found in compatibility database
WARNING:

Software sets are identified as compatible
Verifying image type compatibility
Checking IPC compatibility with running software
Checking candidate package set infrastructure compatibility
Checking infrastructure compatibility with running software
Checking package specific compatibility
Finished compatibility testing

--- Starting impact testing ---
Checking operational impact of change
Finished impact testing
```

```
--- Starting list of software package changes ---
No old package files removed
New files list:
  Added asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg
  Added asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg
Finished list of software package changes

--- Starting commit of software changes ---
Updating provisioning rollback files
Creating pending provisioning file
Committing provisioning file
Finished commit of software changes

--- Starting analysis of software changes ---
Finished analysis of software changes

--- Starting update running software ---
Blocking peer synchronization of operating information
Creating the command set placeholder directory
  Finding latest command set
  Finding latest command shortlist lookup file
  Finding latest command shortlist file
  Assembling CLI output libraries
  Assembling CLI input libraries
  Assembling Dynamic configuration files
  Applying interim IPC and database definitions
  Replacing running software
  Replacing CLI software
  Restarting software

Generating software version information
  Notifying running software of updates
  Unblocking peer synchronization of operating information
Unmounting old packages
Cleaning temporary installation files
  Finished update running software

SUCCESS: Finished installing software.


*Aug  4 19:21:45.424 IST: %IOSXE_OIR-6-ONLINECARD: Card (cc) online in slot 2
*Aug  4 19:21:48.382 IST: %IOSXE_OIR-6-INSSPA: SPA inserted in subslot 2/0
*Aug  4 19:21:48.733 IST: %IOSXE_OIR-6-INSSPA: SPA inserted in subslot 2/1
*Aug  4 19:21:49.083 IST: %IOSXE_OIR-6-INSSPA: SPA inserted in subslot 2/2
*Aug  4 19:21:49.430 IST: %IOSXE_OIR-6-INSSPA: SPA inserted in subslot 2/3
*Aug  4 19:21:58.121 IST: %LINK-3-UPDOWN: SIP2/0: Interface EOBC2/1, changed state to up
*Aug  4 19:22:02.302 IST: %SPA_OIR-6-ONLINECARD: SPA (SPA-1X10GE-L-V2) online in subslot
2/0

*Aug  4 19:22:02.282 IST: %TRANSCEIVER-6-INSERTED: SIP2/0: transceiver module inserted in
TenGigabitEthernet2/0/0
*Aug  4 19:22:02.518 IST: %LINK-3-UPDOWN: SIP2/1: Interface EOBC2/1, changed state to up
*Aug  4 19:22:06.113 IST: %SPA_OIR-6-ONLINECARD: SPA (SPA-1X10GE-L-V2) online in subslot
2/1
*Aug  4 19:22:06.082 IST: %TRANSCEIVER-6-INSERTED: SIP2/1: transceiver module inserted in
TenGigabitEthernet2/1/0
*Aug  4 19:22:08.080 IST: %LINK-3-UPDOWN: SIP2/2: Interface EOBC2/1, changed state to up
*Aug  4 19:22:11.627 IST: %SPA_OIR-6-ONLINECARD: SPA (SPA-1X10GE-L-V2) online in subslot
2/2
*Aug  4 19:22:11.616 IST: %TRANSCEIVER-6-INSERTED: SIP2/2: transceiver module inserted in
TenGigabitEthernet2/2/0
*Aug  4 19:22:12.523 IST: %LINK-3-UPDOWN: SIP2/3: Interface EOBC2/1, changed state to up
```

```
*Aug  4 19:22:16.657 IST: %SPA_OIR-6-ONLINECARD: SPA (SPA-1X10GE-L-V2) online in subslot
2/3
*Aug  4 19:22:16.648 IST: %TRANSCEIVER-6-INSERTED: SIP2/3: transceiver module inserted in
TenGigabitEthernet2/3/0


Router# issu commitversion
--- Starting local lock acquisition on R0 ---
Finished local lock acquisition on R0

--- Starting installation changes ---
Cancelling rollback timer
Finished installation changes

SUCCESS: Installation changes committed


Router# issu loadversion rp 0 file
bootflash:Active_Dir/asr1000rp2-{elcbase,elcspa}*03.13.00.S.154-3.S-ext*.pkg slot 4
--- Starting local lock acquisition on R0 ---
Finished local lock acquisition on R0

--- Starting installation state synchronization ---
Finished installation state synchronization

--- Starting file path checking ---
Finished file path checking

--- Starting image file verification ---
Checking image file names
Locating image files and validating name syntax
  Found asr1000rp2-elcbase.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg
Verifying image file locations
Inspecting image file types
Processing image file constraints
Creating candidate provisioning file
Finished image file verification

--- Starting candidate package set construction ---
Verifying existing software set
Processing candidate provisioning file
Constructing working set for candidate package set
Constructing working set for running package set
Checking command output
Constructing merge of running and candidate packages
Checking if resulting candidate package set would be complete
Finished candidate package set construction

--- Starting compatibility testing ---
Determining whether candidate package set is compatible

WARNING:
WARNING: Candidate software combination not found in compatibility database
WARNING:

Determining whether installation is valid
Creating matrix_file by locate_latest_matrix_file  /tmp/issu/provision/sw

WARNING:
WARNING: Candidate software combination not found in compatibility database
WARNING:
```

```
WARNING:
WARNING: Candidate software combination not found in compatibility database
WARNING:

Software sets are identified as compatible
Verifying image type compatibility
Checking IPC compatibility with running software
Checking candidate package set infrastructure compatibility
Checking infrastructure compatibility with running software
Checking package specific compatibility
Finished compatibility testing

--- Starting impact testing ---
Checking operational impact of change
Finished impact testing

--- Starting list of software package changes ---
No old package files removed
New files list:
  Added asr1000rp2-elcbase.03.13.00.S.154-3.S-ext.pkg
  Added asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg
Finished list of software package changes

--- Starting commit of software changes ---
Updating provisioning rollback files
Creating pending provisioning file
Committing provisioning file
Finished commit of software changes

--- Starting analysis of software changes ---
Finished analysis of software changes

--- Starting update running software ---
Blocking peer synchronization of operating information
Creating the command set placeholder directory
  Finding latest command set
  Finding latest command shortlist lookup file
  Finding latest command shortlist file
  Assembling CLI output libraries
  Assembling CLI input libraries
  Assembling Dynamic configuration files
  Applying interim IPC and database definitions
  Replacing running software
  Replacing CLI software
  Restarting software
  Applying final IPC and database definitions
  *Aug 4 19:22:05.767 IST: %IOSXE_OIR-6-OFFLINECARD: Card (cc) offline in slot 4
  *Aug 4 19:22:05.770 IST: %IOSXE_OIR-6-REMSPA: SPA removed from subslot 4/0, interfaces
  disabled
  Generating software version information
  Notifying running software of updates
  Unblocking peer synchronization of operating information
Unmounting old packages
Cleaning temporary installation files
  Finished update running software

SUCCESS: Finished installing software.

*Aug 4 19:24:05.041 IST: %IOSXE_OIR-6-ONLINECARD: Card (cc) online in slot 4
*Aug 4 19:24:14.250 IST: %IOSXE_OIR-6-INSSPA: SPA inserted in subslot 4/0
*Aug 4 19:24:34.196 IST: %LINK-3-UPDOWN: SIP4/0: Interface EOBC0/1, changed state to up
*Aug 4 19:25:27.923 IST: %LINK-3-UPDOWN: Interface GigabitEthernet4/0/0, changed state to
down
```

```
*Aug 4 19:25:30.497 IST: %LINK-3-UPDOWN: Interface GigabitEthernet4/0/1, changed state to
down
*Aug 4 19:25:40.915 IST: %SPA_OIR-6-ONLINECARD: SPA (BUILT-IN-2T+20X1GE) online in subslot
4/0
*Aug 4 19:25:53.574 IST: %LINK-3-UPDOWN: Interface GigabitEthernet4/0/0, changed state to
up
*Aug 4 19:25:53.582 IST: %LINK-3-UPDOWN: Interface GigabitEthernet4/0/1, changed state to
up

Router# issu commitversion
--- Starting local lock acquisition on R0 ---
Finished local lock acquisition on R0

--- Starting installation changes ---
Cancelling rollback timer
Finished installation changes

SUCCESS: Installation changes committed


Router# issu loadversion rp 0 file
bootflash:Active_Dir/asr1000rp2-esp*03.13.00.S.154-3.S-ext*.pkg slot 1
--- Starting local lock acquisition on R0 ---
Finished local lock acquisition on R0

--- Starting installation state synchronization ---
Finished installation state synchronization

--- Starting file path checking ---
Finished file path checking

--- Starting image file verification ---
Checking image file names
Locating image files and validating name syntax
  Found asr1000rp2-espbase.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-espx86base.03.13.00.S.154-3.S-ext.pkg
Verifying image file locations
Inspecting image file types
Processing image file constraints
Creating candidate provisioning file
Finished image file verification

--- Starting candidate package set construction ---
Verifying existing software set
Processing candidate provisioning file
Constructing working set for candidate package set
Constructing working set for running package set
Checking command output
Constructing merge of running and candidate packages
Checking if resulting candidate package set would be complete
Finished candidate package set construction

--- Starting compatibility testing ---
Determining whether candidate package set is compatible

WARNING:
WARNING: Candidate software combination not found in compatibility database
WARNING:

Determining whether installation is valid
Creating matrix_file by locate_latest_matrix_file  /tmp/issu/provision/sw

WARNING:
WARNING: Candidate software combination not found in compatibility database
```

```
WARNING:


WARNING:
WARNING: Candidate software combination not found in compatibility database
WARNING:

Software sets are identified as compatible
Verifying image type compatibility
Checking IPC compatibility with running software
Checking candidate package set infrastructure compatibility
Checking infrastructure compatibility with running software
Checking package specific compatibility
Finished compatibility testing

--- Starting impact testing ---
Checking operational impact of change
Finished impact testing

--- Starting list of software package changes ---
No old package files removed
New files list:
  Added asr1000rp2-espbase.03.13.00.S.154-3.S-ext.pkg
  Added asr1000rp2-espx86base.03.13.00.S.154-3.S-ext.pkg
Finished list of software package changes

--- Starting commit of software changes ---
Updating provisioning rollback files
Creating pending provisioning file
Committing provisioning file
Finished commit of software changes

--- Starting analysis of software changes ---
Finished analysis of software changes

--- Starting update running software ---
Blocking peer synchronization of operating information
Creating the command set placeholder directory
  Finding latest command set
  Finding latest command shortlist lookup file
  Finding latest command shortlist file
  Assembling CLI output libraries
  Assembling CLI input libraries
  Assembling Dynamic configuration files
  Applying interim IPC and database definitions
  Replacing running software
  Replacing CLI software
  Restarting software
    Restarting ESP1
  Applying final IPC and database definitions

Aug  4 19:29:16.751 IST: %IOSXE_OIR-6-OFFLINECARD: Card (fp) offline in slot F1
*Aug  4 19:29:18.172 IST: %CMRP-6-FP_HA_STATUS: R0/0: cmand:  F0 redundancy state is
Active with no Standby  Generating software version information
  Notifying running software of updates
  Unblocking peer synchronization of operating information
Unmounting old packages
Cleaning temporary installation files
  Finished update running software

SUCCESS: Finished installing software.
```

```
*Aug  4 19:30:50.972 IST: %CPPHA-7-START: F1: cpp_ha:  CPP 0 preparing image
/tmp/sw/fp/1/0/fpx86/mount/usr/cpp/bin/qfp-ucode-esp40
*Aug  4 19:30:51.362 IST: %CPPHA-7-START: F1: cpp_ha:  CPP 0 startup init image
/tmp/sw/fp/1/0/fpx86/mount/usr/cpp/bin/qfp-ucode-esp40
*Aug  4 19:30:53.088 IST: %IOSXE_OIR-6-ONLINECARD: Card (fp) online in slot F1


Router# issu commitversion
--- Starting local lock acquisition on R0 ---
Finished local lock acquisition on R0

--- Starting installation changes ---
Cancelling rollback timer
Finished installation changes

SUCCESS: Installation changes committed


Router# issu loadversion rp 0 file
bootflash:Active_Dir/asr1000rp2-esp*03.13.00.S.154-3.S-ext*.pkg slot 0
--- Starting local lock acquisition on R0 ---
Finished local lock acquisition on R0

--- Starting installation state synchronization ---
Finished installation state synchronization

--- Starting file path checking ---
Finished file path checking

--- Starting image file verification ---
Checking image file names
Locating image files and validating name syntax
  Found asr1000rp2-espbase.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-espx86base.03.13.00.S.154-3.S-ext.pkg
Verifying image file locations
Inspecting image file types
Processing image file constraints
Creating candidate provisioning file

Aug  4 19:31:14.730 IST: %CPPHA-7-START: F1: cpp_ha:  CPP 0 running init image
/tmp/sw/fp/1/0/fpx86/mount/usr/cpp/bin/qfp-ucode-esp40
*Aug  4 19:31:15.079 IST: %CPPHA-7-READY: F1: cpp_ha:  CPP 0 loading and initialization
completeFinished image file verification

--- Starting candidate package set construction ---
Verifying existing software set

*Aug  4 19:31:15.309 IST: %IOSXE-6-PLATFORM: F1: cpp_cp: Process
CPP_PFILTER_EA_EVENT__API_CALL__REGISTERProcessing candidate provisioning file

*Aug  4 19:31:18.010 IST: %CMRP-6-FP_HA_STATUS: R0/0: cmand:  F0 redundancy state is
Active with ready StandbyConstructing working set for candidate package


setConstructing working set for running package set
Checking command output
Constructing merge of running and candidate packages
Checking if resulting candidate package set would be complete
Finished candidate package set construction

--- Starting compatibility testing ---
Determining whether candidate package set is compatible

WARNING:
```

```
WARNING: Candidate software combination not found in compatibility database
WARNING:

Determining whether installation is valid
Creating matrix_file by locate_latest_matrix_file  /tmp/issu/provision/sw
Software sets are identified as compatible
Verifying image type compatibility
Checking IPC compatibility with running software
Checking candidate package set infrastructure compatibility
Checking infrastructure compatibility with running software
Checking package specific compatibility
Finished compatibility testing

--- Starting impact testing ---
Checking operational impact of change
Finished impact testing

--- Starting list of software package changes ---
Old files list:
  Removed asr1000rp2-espbase.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-espx86base.03.12.01.S.154-2.S.pkg
No new package files added
Finished list of software package changes

--- Starting commit of software changes ---
Updating provisioning rollback files
Creating pending provisioning file
Committing provisioning file
Finished commit of software changes

--- Starting analysis of software changes ---
Finished analysis of software changes

--- Starting update running software ---
Blocking peer synchronization of operating information
Creating the command set placeholder directory
  Finding latest command set
  Finding latest command shortlist lookup file
  Finding latest command shortlist file
  Assembling CLI output libraries
  Assembling CLI input libraries
  Assembling Dynamic configuration files
  Applying interim IPC and database definitions
  Replacing running software
  Replacing CLI software
  Restarting software
    Restarting ESP0
  Applying final IPC and database definitions

*Aug  4 19:32:46.187 IST: %IOSXE_OIR-6-OFFLINECARD: Card (fp) offline in slot F0
*Aug  4 19:32:46.539 IST: %CMRP-6-FP_HA_STATUS: R0/0: cmand:  F1 redundancy state is
Active  Generating software version information
  Notifying running software of updates
  Unblocking peer synchronization of operating information
Unmounting old packages
Cleaning temporary installation files
  Finished update running software

SUCCESS: Finished installing software.


*Aug  4 19:34:19.748 IST: %CPPHA-7-START: F0: cpp_ha:  CPP 0 preparing image
/tmp/sw/fp/0/0/fpx86/mount/usr/cpp/bin/qfp-ucode-esp40
```

```
*Aug  4 19:34:20.139 IST: %CPPHA-7-START: F0: cpp_ha:  CPP 0 startup init image
/tmp/sw/fp/0/0/fpx86/mount/usr/cpp/bin/qfp-ucode-esp40
*Aug  4 19:34:21.858 IST: %IOSXE_OIR-6-ONLINECARD: Card (fp) online in slot F0
*Aug  4 19:34:43.609 IST: %CPPHA-7-START: F0: cpp_ha:  CPP 0 running init image
/tmp/sw/fp/0/0/fpx86/mount/usr/cpp/bin/qfp-ucode-esp40
*Aug  4 19:34:43.958 IST: %CPPHA-7-READY: F0: cpp_ha:  CPP 0 loading and initialization
complete
*Aug  4 19:34:44.190 IST: %IOSXE-6-PLATFORM: F0: cpp_cp: Process
CPP_PFILTER_EA_EVENT__API_CALL__REGISTER
*Aug  4 19:34:46.890 IST: %CMRP-6-FP_HA_STATUS: R0/0: cmand:  F0 redundancy state is
Standby
```

```
Router# issu commitversion
--- Starting local lock acquisition on R0 ---
Finished local lock acquisition on R0

--- Starting installation changes ---
Cancelling rollback timer
Finished installation changes

SUCCESS: Installation changes committed
```

```
Router# issu loadversion rp 0 file
bootflash:Active_Dir/asr1000rp2*03.13.00.S.154-3.S-ext*.pkg force
--- Starting local lock acquisition on R0 ---
Finished local lock acquisition on R0
s
--- Starting installation state synchronization ---
Finished installation state synchronization

--- Starting file path checking ---
Finished file path checking

--- Starting image file verification ---
Checking image file names
Locating image files and validating name syntax
  Found asr1000rp2-elcbase.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-espbase.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-espx86base.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-rpaccess.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-rpbase.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-rpcontrol.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-rpios-adventerprisek9.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg
Verifying image file locations
Inspecting image file types
    WARNING: In-service installation of IOSD package
    WARNING: requires software redundancy on target RP
    WARNING: or on-reboot parameter
    WARNING: Automatically setting the on-reboot flag
    WARNING: In-service installation of RP Base package
    WARNING: requires software reboot of target RP
Processing image file constraints
Creating candidate provisioning file
Finished image file verification

--- Starting candidate package set construction ---
Verifying existing software set
Processing candidate provisioning file
```

```
Constructing working set for candidate package set
Constructing working set for running package set
Checking command output
Constructing merge of running and candidate packages
Checking if resulting candidate package set would be complete
Finished candidate package set construction

--- Starting compatibility testing ---
Determining whether candidate package set is compatible
Determining whether installation is valid
Determining whether installation is valid ... skipped
Verifying image type compatibility
Checking IPC compatibility for candidate software
Checking candidate package set infrastructure compatibility
Checking infrastructure compatibility with running software
Checking infrastructure compatibility with running software ... skipped
Checking package specific compatibility
Finished compatibility testing

--- Starting list of software package changes ---
Old files list:
  Removed asr1000rp2-elcbase.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-elcspa.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-rpaccess.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-rpbase.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-rpcontrol.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-rpios-adventerprisek9.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-sipbase.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-sipspa.03.12.01.S.154-2.S.pkg
New files list:
  Added asr1000rp2-rpaccess.03.13.00.S.154-3.S-ext.pkg
  Added asr1000rp2-rpbase.03.13.00.S.154-3.S-ext.pkg
  Added asr1000rp2-rpcontrol.03.13.00.S.154-3.S-ext.pkg
  Added asr1000rp2-rpios-adventerprisek9.03.13.00.S.154-3.S-ext.pkg
Finished list of software package changes

--- Starting commit of software changes ---
Updating provisioning rollback files
Creating pending provisioning file
Committing provisioning file
Finished commit of software changes

SUCCESS: Software provisioned.  New software will load on reboot.


Router# show  platform
Chassis type: ASR1013

Slot      Type                State                 Insert time (ago)
--------- ------------------- --------------------- -----------------
 2        ASR1000-SIP40        ok                    1d04h
 2/0      SPA-1X10GE-L-V2      ok                    1d04h
 2/1      SPA-1X10GE-L-V2      ok                    1d04h
 2/2      SPA-1X10GE-L-V2      ok                    1d04h
 2/3      SPA-1X10GE-L-V2      ok                    1d04h
 4        ASR1000-2T+20X1GE    ok                    1d04h
 4/0      BUILT-IN-2T+20X1GE   ok                    1d04h
 R0       ASR1000-RP2          ok, active            1d04h
 R1       ASR1000-RP2          ok, standby           1d04h
 F0       ASR1000-ESP100       ok, standby           1d04h
 F1       ASR1000-ESP100       ok, active            1d04h
 P0       ASR1013-PWR-AC       ok                    1d04h
 P1       ASR1013-PWR-AC       ok                    1d04h
 P2       ASR1013-PWR-AC       ok                    1d04h
```

```
 P3         ASR1013-PWR-AC         ps, fail              1d04h

Slot      CPLD Version        Firmware Version
--------- ------------------- ----------------------------------------
 2         00200800            15.3(3r)S
 4         00200800            15.2(1r)S
 R0        10021901            15.3(3r)S
 R1        10021901            15.3(3r)S
 F0        12071700            15.3(3r)S
 F1        12071700            15.3(3r)S


Router# show version R0 provisioned
Package: Provisioning File, version: n/a, status: active
  File: bootflash:Active_Dir/packages.conf, on: RP0
  Built: n/a, by: n/a
  File SHA1 checksum: c79075780592aec1312725f4a2357a034fda2d3b

Package: rpbase, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-rpbase.03.13.00.S.154-3.S-ext.pkg, on: RP0
  Built: 2013-07-25_22.55, by: mcpre
  File SHA1 checksum: 4f655c54bb95b4dfa24a0d25ebf97cf8527c69e9

Package: rpcontrol, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-rpcontrol.03.13.00.S.154-3.S-ext.pkg, on: RP0/0
  Built: 2013-07-25_22.55, by: mcpre
  File SHA1 checksum: 8a0a45ea5c7a656c0eef6726174461584f182c78

Package: rpios-adventerprisek9, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-rpios-adventerprisek9.03.13.00.S.154-3.S-ext.pkg,
on: RP0/0
  Built: 2013-07-25_23.00, by: mcpre
  File SHA1 checksum: 85e9eab826bff2194ef568a56c76453625383ad2

Package: rpaccess, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-rpaccess.03.13.00.S.154-3.S-ext.pkg, on: RP0/0
  Built: 2013-07-25_22.55, by: mcpre
  File SHA1 checksum: a360dff0fd76a9b1ae67cda9116c97b62f25ab09

Package: rpcontrol, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-rpcontrol.03.13.00.S.154-3.S-ext.pkg, on: RP0/1
  Built: 2013-07-25_22.55, by: mcpre
  File SHA1 checksum: 8a0a45ea5c7a656c0eef6726174461584f182c78

Package: rpios-adventerprisek9, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-rpios-adventerprisek9.03.13.00.S.154-3.S-ext.pkg,
on: RP0/1
  Built: 2013-07-25_23.00, by: mcpre
  File SHA1 checksum: 85e9eab826bff2194ef568a56c76453625383ad2

Package: rpaccess, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-rpaccess.03.13.00.S.154-3.S-ext.pkg, on: RP0/1
  Built: 2013-07-25_22.55, by: mcpre
  File SHA1 checksum: a360dff0fd76a9b1ae67cda9116c97b62f25ab09

Package: rpbase, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-rpbase.03.13.00.S.154-3.S-ext.pkg, on: RP1
  Built: 2013-07-25_22.55, by: mcpre
  File SHA1 checksum: 4f655c54bb95b4dfa24a0d25ebf97cf8527c69e9

Package: rpcontrol, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-rpcontrol.03.13.00.S.154-3.S-ext.pkg, on: RP1/0
  Built: 2013-07-25_22.55, by: mcpre
  File SHA1 checksum: 8a0a45ea5c7a656c0eef6726174461584f182c78
```

```
<some output removed for brevity>

Router# show version R0 provisioned
Package: Provisioning File, version: n/a, status: active
  File: bootflash:Active_Dir/packages.conf, on: RP0
  Built: n/a, by: n/a
  File SHA1 checksum: c79075780592aec1312725f4a2357a034fda2d3b

Package: rpbase, version: 03.12.01.S.154-2.S, status: active
  File: bootflash:Active_Dir/asr1000rp2-rpbase.03.12.01.S.154-2.S.pkg, on: RP0
  Built: 2013-03-25_18.48, by: mcpre
  File SHA1 checksum: 3a9675142898cfac350d4e42f0e37bd9f4e48538

Package: rpcontrol, version: 03.12.01.S.154-2.S, status: active
  File: bootflash:Active_Dir/asr1000rp2-rpcontrol.03.12.01.S.154-2.S.pkg, on: RP0/0
  Built: 2013-03-25_18.48, by: mcpre
  File SHA1 checksum: 87b11f863f67fdf2610ee0769b929baab4c3efad

Package: rpios-adventerprisek9, version: 03.12.01.S.154-2.S, status: active
  File: bootflash:Active_Dir/asr1000rp2-rpios-adventerprisek9.03.12.01.S.154-2.S.pkg, on:
RP0/0
  Built: 2013-03-25_18.51, by: mcpre
  File SHA1 checksum: b487136319da0a327844d353c77e533c53c56053

Package: rpaccess, version: 03.12.01.S.154-2.S, status: active
  File: bootflash:Active_Dir/asr1000rp2-rpaccess.03.12.01.S.154-2.S.pkg, on: RP0/0
  Built: 2013-03-25_18.48, by: mcpre
  File SHA1 checksum: 032bea36f74b19977b363243c99f02413b54104d


<some output removed for brevity>
Router# redundancy force-switchover
Proceed with switchover to standby RP? [confirm]
<output removed for brevity>


Router# request platform software package clean
Cleaning up unnecessary package files
No path specified, will use booted
path bootflash:Active_Dir/packages.conf
Cleaning bootflash:Active_Dir
Scanning boot directory for packages ... done.
Preparing packages list to delete ... asr1000rp2-elcbase.03.13.00.S.154-3.S-ext.pkg
      File is in use, will not delete...asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg
      File is in use, will not delete...asr1000rp2-espbase.03.13.00.S.154-3.S-ext.pkg
      File is in use, will not delete...asr1000rp2-espx86base.03.13.00.S.154-3.S-ext.pkg
      File is in use, will not delete...asr1000rp2-rpaccess.03.13.00.S.154-3.S-ext.pkg
      File is in use, will not delete...asr1000rp2-rpbase.03.13.00.S.154-3.S-ext.pkg
      File is in use, will not delete...asr1000rp2-rpcontrol.03.13.00.S.154-3.S-ext.pkg
      File is in use, will not delete...asr1000rp2-rpios-adventerprisek9.03.10.00.S.153
-3.S-ext.pkg
      File is in use, will not delete...asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg
      File is in use, will not delete...asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg
      File is in use, will not delete...packages.conf
      File is in use, will not delete... done.

Files that will be deleted:  asr1000rp2-elcbase.03.12.01.S.154-2.S.pkg
  asr1000rp2-elcspa.03.12.01.S.154-2.S.pkg
  asr1000rp2-espbase.03.12.01.S.154-2.S.pkg
  asr1000rp2-espx86base.03.12.01.S.154-2.S.pkg
  asr1000rp2-packages-adventerprisek9.03.12.01.S.154-2.S.conf
  asr1000rp2-rpaccess.03.12.01.S.154-2.S.pkg
```

```
  asr1000rp2-rpbase.03.12.01.S.154-2.S.pkg
  asr1000rp2-rpcontrol.03.12.01.S.154-2.S.pkg
  asr1000rp2-rpios-adventerprisek9.03.12.01.S.154-2.S.pkg
  asr1000rp2-sipbase.03.12.01.S.154-2.S.pkg
  asr1000rp2-sipspa.03.12.01.S.154-2.S.pkg
  packages.conf.00-

Do you want to proceed? [confirm]y
  Deleting file bootflash:Active_Dir/asr1000rp2-elcbase.03.12.01.S.154-2.S.pkg ... done.
  Deleting file bootflash:Active_Dir/asr1000rp2-elcspa.03.12.01.S.154-2.S.pkg ... done.
  Deleting file bootflash:Active_Dir/asr1000rp2-espbase.03.12.01.S.154-2.S.pkg ... done.
  Deleting file bootflash:Active_Dir/asr1000rp2-espx86base.03.12.01.S.154-2.S.pkg... done.
  Deleting file
bootflash:Active_Dir/asr1000rp2-packages-adventerprisek9.03.12.01.S.154-2.S.conf ... done.
  Deleting file bootflash:Active_Dir/asr1000rp2-rpaccess.03.12.01.S.154-2.S.pkg ... done.
  Deleting file bootflash:Active_Dir/asr1000rp2-rpbase.03.12.01.S.154-2.S.pkg ... done.
  Deleting file bootflash:Active_Dir/asr1000rp2-rpcontrol.03.12.01.S.154-2.S.pkg ... done.
  Deleting file
bootflash:Active_Dir/asr1000rp2-rpios-adventerprisek9.03.12.01.S.154-2.S.pkg ... done.
  Deleting file bootflash:Active_Dir/asr1000rp2-sipbase.03.12.01.S.154-2.S.pkg ... done.
  Deleting file bootflash:Active_Dir/asr1000rp2-sipspa.03.12.01.S.154-2.S.pkg ... done.
  Deleting file bootflash:Active_Dir/packages.conf.00- ... done.
SUCCESS: Files deleted.
```

Using ISSU to Upgrade Subpackages on a Cisco ASR 1006 Router or ASR 1013 Router (request platform command set)

This procedure can only be performed if the current ASR 1006 router or ASR 1013 router has two active RPs and both RPs are running subpackages.

To perform an ISSU upgrade using subpackages on a Cisco ASR 1006 router or ASR 1013 router with a dual RP setup using the **request platform** command set, follow the following instructions.

SUMMARY STEPS

1. **show version**

   show version *active-rp* **installed**

   show version *standby-rp* **installed**

   dir *filesystem*:<*directory*>

   show platform
2. **mkdir** *URL-to-directory-name*
3. **ip tftp** *source-interface gigabitethernet port*
4. **copy tftp:** *URL-to-target-location*
5. **request platform software package expand file** *URL-to-consolidated-package*
6. **dir** *URL-to-consolidated-package*
7. **copy** *file-system*:**asr1000rp2-espbase.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*

   copy *file-system*:**asr1000rp2-espx86base.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*

   copy *file-system*:**asr1000rp2-rpaccess.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*

   copy *file-system*:**asr1000rp2-rpbase.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*

   copy *file-system*:**asr1000rp2-rpcontrol.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*

   copy *file-system*:**asr1000rp2-rpios.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*

copy *file-system*:**asr1000rp2-sipbase.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*

copy *file-system*:**asr1000rp2-sipspa.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*

opy *file-system*:**asr1000rp2-elcbase.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*

copy *file-system*:**asr1000rp2-elcspa.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*

Note In step 7, each individual subpackage that was extracted in step 5 is copied to the directory where the subpackages that are currently running the active RP are stored.

8. **copy** *file-system*:**asr1000rp2-espbase.***version*.**pkg** *URL-to-directory-of-sub-packages-standby-RP*

   copy *file-system*:**asr1000rp2-espx86base.***version*.**pkg** *URL-to-directory-of-sub-packages-standby-RP*

   copy *file-system*:**asr1000rp2-rpaccess.***version*.**pkg** *URL-to-directory-of-sub-packages-standby-RP*

   copy *file-system*:**asr1000rp2-rpbase.***version*.**pkg** *URL-to-directory-of-sub-packages-standby-RP*

   copy *file-system*:**asr1000rp2-rpcontrol.***version*.**pkg** *URL-to-directory-of-sub-packages-standby-RP*

   copy *file-system*:**asr1000rp2-rpios.***version*.**pkg** *URL-to-directory-of-sub-packages-standby-RP*

   copy *file-system*:**asr1000rp2-sipbase.***version*.**pkg** *URL-to-directory-of-sub-packages-standby-RP*

   copy *file-system*:**asr1000rp2-sipspa.***version*.**pkg** *URL-to-directory-of-sub-packages-standby-RP*

   copy *file-system*:**asr1000rp2-elcbase.***version*.**pkg** *URL-to-directory-of-sub-packages-standby-RP*

   copy *file-system*:**asr1000rp2-elcspa.***version*.**pkg** *URL-to-directory-of-sub-packages-standby-RP*

Note In step 8, each individual subpackage that was extracted in step 5 is copied to the directory where the subpackages that are currently running the standby RP are stored.

9. **request platform software package install rp** *standby-RP* **file** *URL-to-standby-file-system*:**asr1000rp***version*.**pkg force**
10. **hw-module slot** *standby-RP* **reload**
11. **request platform software package install rp** *active-RP* **file** *URL-to-active-file-system*:*image* **slot** *SIP-slot-number* **force**

    Repeat the step 11, for each available SIP installed in the router before moving onto the next step.
12. **request platform software package install rp** *active-RP* **file** *URL-to-active-file-system*:**asr1000rp2-{elcbase,elcspa}***version*.**pkg slot** *ELC-slot-number* **force**

    Repeat the step 12, for each available ELC installed in the router before moving onto the next step.
13. **request platform software package install rp** *active-RP* **file** *URL-to-active-file-system*:**asr1000rp2-esp***version*.**pkg slot** *standby-ESP-slot*

    request platform software package install rp *active-RP* **file** *URL-to-active-file-system*:**asr1000rp2-esp***version*.**pkg slot** *active-ESP-slot*
14. **request platform software package install rp** *active-RP* **file** *URL-to-active-file-system*:**asr1000rp***version*.**pkg force**
15. **show version** *active-RP* **provisioned**

    show version *active-RP* **installed**

16. **redundancy force-switchover**
17. request platform software package clean

DETAILED STEPS

	Command or Action	Purpose

Step 1	**show version**
	show version *active-rp* **installed**
	show version *standby-rp* **installed**
	dir *filesystem*:*<directory>*
	show platform
	
	Example:
	`Router# show version`
	`Router# show version r0 installed`
	`Router# show version r1 installed`
	`Router# dir bootflash:`
	`Router# show platform`
		(Optional) Use the following commands to confirm the current router configuration, as follows:
		• **show version** and **show version** *active-rp* **installed**—Verify the running version of the Cisco IOS XE software on the router, and which file was used to boot the router, and where that file is stored.
		• **dir**—Confirm that the files that were used to boot the router are located in the directory.
		• **show platform**—Confirm the current status of the active and standby RPs.

Step 2	**mkdir** *URL-to-directory-name* **Example:** `Router# mkdir usb0:221subs`	Creates a directory to store the consolidated package and subpackages. This directory must be created in most cases because the consolidated packages and subpackages have to be separated from the subpackages that booted the router at this step of the procedure.
Step 3	**ip tftp source-interface gigabitethernet** *port* **Example:** `Router(config)# ip tftp source-interface gigabitethernet 0`	Specifies the Gigabit Ethernet TFTP source-interface to be configured: *slot/port*—Specifies the location of the TFTP source-interface. **Note** To copy a file using TFTP through the Management Ethernet interface, the **ip tftp source-interface GigabitEthernet 0** command must be entered before entering the **copy tftp** command.
Step 4	**copy tftp:** *URL-to-target-location* **Example:** `Router# copy tftp: usb0:221subs`	Copies the consolidated package file into the directory created in Step 2. The consolidated package in this step should not be copied into the same directory where the subpackages that are currently running your router are stored (the directory containing the packages.conf provisioning file from which the router was booted). **Tip** It is recommended that you copy the package onto a usb: or harddisk: file system for space considerations when performing this step of the procedure.

	Command or Action	Purpose
Step 5	**request platform software package expand file** *URL-to-consolidated-package* **Example:** `Router# request platform software package expand file usb0:221subs/asr1000rp2-adventerprisek9.03.13.00.S.154-3.S-ext.bin`	Extracts the subpackages out of the consolidated package file into the temporary directory. **Note** Take extra care to extract the subpackages to a temporary subdirectory and do not delete any of the files currently running the router at this point of the procedure. To erase the files that were running on the router before the ISSU upgrade, enter the **request platform software package clean** command after the ISSU upgrade has been completed.
Step 6	**dir** *target-URL* **Example:** `Router# dir usb0:221subs`	(Optional) Displays the directory to confirm that the files were extracted.

	Command or Action	Purpose
Step 7	**copy** *file-system*:**asr1000rp2-espbase.**_version_.**pkg** *URL-to-directory-of-sub-packages-active-RP* **copy** *file-system*:**asr1000rp2-espx86base.**_version_.**pkg** *URL-to-directory-of-sub-packages-active-RP* **copy** *file-system*:**asr1000rp2-rpaccess.**_version_.**pkg** *URL-to-directory-of-sub-packages-active-RP* **copy** *file-system*:**asr1000rp2-rpbase.**_version_.**pkg** *URL-to-directory-of-sub-packages-active-RP* **copy** *file-system*:**asr1000rp2-rpcontrol.**_version_.**pkg** *URL-to-directory-of-sub-packages-active-RP* **copy** *file-system*:**asr1000rp2-rpios.**_version_.**pkg** *URL-to-directory-of-sub-packages-active-RP* **copy** *file-system*:**asr1000rp2-sipbase.**_version_.**pkg** *URL-to-directory-of-sub-packages-active-RP* **copy** *file-system*:**asr1000rp2-sipspa.**_version_.**pkg** *URL-to-directory-of-sub-packages-active-RP* **copy** *file-system*:**asr1000rp2-elcbase.**_version_.**pkg** *URL-to-directory-of-sub-packages-active-RP* **copy** *file-system*:**asr1000rp2-elcspa.**_version_.**pkg** *URL-to-directory-of-sub-packages-active-RP* **Example:** Router# **copy usb0:221subs/asr1000rp2-espbase.03.13.00.S.154-3.S-ext.pkg bootflash:** Router# **copy usb0:221subs/asr1000rp2-espx86base.03.13.00.S.154-3.S-ext.pkg bootflash:** Router# **copy usb0:221subs/asr1000rp2-rpaccess.03.13.00.S.154-3.S-ext.pkg bootflash:** Router# **copy usb0:221subs/asr1000rp2-rpbase.03.13.00.S.154-3.S-ext.pkg bootflash:** Router# **copy usb0:221subs/asr1000rp2-rpcontrol.03.13.00.S.154-3.S-ext.pkg bootflash:** Router# **copy usb0:221subs/asr1000rp2-rpios-adventerprisek9.03.13.00.S.154-3.S-ext.pkg bootflash:** Router# **copy usb0:221subs/asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg bootflash:** Router# **copy usb0:221subs/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg bootflash:** Router# **copy usb0:221subs/asr1000rp2-elcbase.03.13.00.S.154-3.S-ext.pkg bootflash:** Router# **copy usb0:221subs/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg bootflash:**	Copies the subpackages out of the temporary directory into the directory on the router where the subpackages running the active RP are currently stored. **Note** Make sure to copy all the subpackage files extracted in step 5 to the current active packages directory.

Command or Action	Purpose

Step 8 **copy** *file-system*:**asr1000rp2-espbase.**_version_.**pkg** *URL-to-directory-of-sub-packages-standby-RP*
copy *file-system*:**asr1000rp2-espx86base.**_version_.**pkg** *URL-to-directory-of-sub-packages-standby-RP*
copy *file-system*:**asr1000rp2-rpaccess.**_version_.**pkg** *URL-to-directory-of-sub-packages-standby-RP*
copy *file-system*:**asr1000rp2-rpbase.**_version_.**pkg** *URL-to-directory-of-sub-packages-standby-RP*
copy *file-system*:**asr1000rp2-rpcontrol.**_version_.**pkg** *URL-to-directory-of-sub-packages-standby-RP*
copy *file-system*:**asr1000rp2-rpios.**_version_.**pkg** *URL-to-directory-of-sub-packages-standby-RP*
copy *file-system*:**asr1000rp2-sipbase.**_version_.**pkg** *URL-to-directory-of-sub-packages-standby-RP*
copy *file-system*:**asr1000rp2-sipspa.**_version_.**pkg** *URL-to-directory-of-sub-packages-standby-RP*
copy *file-system*:**asr1000rp2-elcbase.**_version_.**pkg** *URL-to-directory-of-sub-packages-standby-RP*
copy *file-system*:**asr1000rp2-elcspa.**_version_.**pkg** *URL-to-directory-of-sub-packages-standby-RP*
 	
Example:
`Router# copy usb0:221subs/asr1000rp2-espbase.03.13.00.S.154-3.S-ext.pkg stby-bootflash:`
 	
`Router# copy usb0:221subs/asr1000rp2-espx86base.03.13.00.S.154-3.S-ext.pkg stby-bootflash:`
 	
`Router# copy usb0:221subs/asr1000rp2-rpaccess.03.13.00.S.154-3.S-ext.pkg stby-bootflash:`
 	
`Router# copy usb0:221subs/asr1000rp2-rpbase.03.13.00.S.154-3.S-ext.pkg stby-bootflash:`
 	
`Router# copy usb0:221subs/asr1000rp2-rpcontrol.03.13.00.S.154-3.S-ext.pkg stby-bootflash:`
 	
`Router# copy usb0:221subs/asr1000rp2-rpios-adventerprisek9.03.13.00.S.154-3.S-ext.pkg stby-bootflash:`
 	
`Router# copy usb0:221subs/asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg stby-bootflash:`
 	
`Router# copy usb0:221subs/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg stby-bootflash:`
 	
`Router# copy usb0:221subs/asr1000rp2-elcbase.03.13.00.S.154-3.S-ext.pkg stby-bootflash:`
 	
`Router# copy usb0:221subs/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg stby-bootflash:`
	Copies the subpackages out of the temporary directory into the directory on the router where the subpackages running the standby RP are currently stored.
	
	Note Make sure to copy all the subpackage files extracted in step 5 to the current active packages directory.

	Command or Action	Purpose
Step 9	**request platform software package install rp** *standby-RP* **file** *target-standbyRP-URL-for-sub-packages***:asr1000rp*** *version****.pkg force** **Example:** `Router# request platform software package install rp 1 file stby-bootflash:asr1000rp*03.13.00.S.154-3.S-ext*.pkg force`	Upgrades the RP subpackages on the standby RP, where the "**rp***" wildcard is specified to capture all of the RP subpackages for the desired upgrade release.
Step 10	**hw-module slot** *standby-RP* **reload** **Example:** `Router# hw-module slot R1 reload`	Reloads the standby RP.
Step 11	**request platform software package install rp** *active-RP* **file** *URL-to-active-file-system***:asr1000rp2-{sipbase,sipspa}*** *version****.pkg slot** *SIP-slot-number* **force** **Example:** `Router# request platform software package install rp 0 file bootflash:asr1000rp2-{sipbase,sipspa}*03.13.00.S.154-3.S-ext*.pkg slot 2 force`	Upgrades the SIP and SPA subpackages for each SIP on the router. **Note** This step must be completed one SIP at a time, and repeated for each SIP installed on the router before performing the next step. **Tip** You can use the **show ip interface brief** command to identify which slots contain SIPs and SPAs. The interfaces with three numbers (in the form *SIP-number*/*SPA-number*/*interface-number*) identify the SIP and SPA locations in the router. **Note** The *pattern* options used in this CLI (**sipbase** and **sipspa**) were introduced in Cisco IOS XE Release 2.1.2 and are not available in previous Cisco IOS XE Releases. See the "ISSU Procedures (Prior to Cisco IOS XE Release 2.1.2)" section on page 5-72 for pre-Cisco IOS XE Release 2.1.2 ISSU upgrade procedures.

	Command or Action	Purpose
Step 12	**request platform software package install rp** *active-RP* **file** *URL-to-active-file-system*:**asr1000rp2-{elcbase,elcspa}***version****.pkg slot** *ELC-slot-number* **force** **Example:** `Router# request platform software package install rp 0 file bootflash:asr1000rp2-{elcbase,elcspa}*03.13.00.S.154-3.S-ext*.pkg slot 4 force`	Upgrades the ELC and SPA subpackages for each ELC on the router. **Note** This step must be completed for one ELC at a time, and repeated for each ELC installed on the router before performing the next step. **Tip** You can use the **show ip interface brief** command to identify which slots contain ELCs and SPAs. The interfaces with three numbers (in the form *ELC-number*/*SPA-number*/*interface-number*) identify the ELC and SPA locations in the router. **Note** The *pattern* options used in this CLI (**elcbase** and **elcspa**) were introduced in Cisco IOS XE Release 3.10S and are not available in previous Cisco IOS XE Releases.

Step 13	**request platform software package install rp** *active-RP* **file** *URL-to-active-file-system*:**asr1000rp2-esp***version****.pkg slot** *standby-ESP-slot*
	
	request platform software package install rp *active-RP* **file** *URL-to-active-file-system*:**asr1000rp2-esp***version****.pkg slot** *active-ESP-slot*
	
	Example:
	`Router# request platform software package install rp 0 file bootflash:asr1000rp2-esp*03.13.00.S.154-3.S-ext*.pkg slot 1`
	`Router# request platform software package install rp 0 file bootflash:asr1000rp2-esp*03.13.00.S.154-3.S-ext*.pkg slot 0`
		Upgrades the ESP Base subpackage on the standby and the active ESPs.
		
		After entering the **issu loadversion rp** command on the active RP, the ESP switchover will occur automatically. Minimal traffic interruption will occur as a result of this switchover.

Step 14	**request platform software package install rp** *active-RP* **file** *URL-to-active-file-system*:**asr1000rp***version****.pkg force** **Example:** `Router# request platform software package install rp 0 file bootflash:asr1000rp2*03.13.00.S.154-3.S-ext*.pkg force`	Upgrades all of the sub-packages on the active RP. **Note** This step is required to ensure that all subpackages on the router were upgraded as part of this procedure, and might upgrade some subpackages that would otherwise be missed in the process.

Step 15	**show version** *active-RP* **provisioned**
	show version *active-RP* **installed**
	
	Example:
	`Router# show version r0 provisioned`
	`Router# show version r0 installed`
		(Optional) Confirms the subpackages are provisioned and installed.

	Command or Action	Purpose
Step 16	**redundancy force-switchover** **Example:** `Router# redundancy force-switchover`	Forces an RP switchover to complete the upgrade.
Step 17	**request platform software package clean**	(Optional) Removes all unused subpackage files from the router.

Examples

This example shows ISSU upgrade using subpackages on a Cisco ASR 1006 router or ASR 1013 router with a dual RP setup:

```
Router# show version
Cisco IOS Software, IOS-XE Software (X86_64_LINUX_IOSD-ADVENTERPRISEK9-M), Version
15.3(2)S, RELEASE SOFTWARE (fc1)
<output removed for brevity>
System image file is "bootflash:Active_Dir/packages.conf"
<output removed for brevity>
cisco ASR1013 (RP2) processor with 4208889K/6147K bytes of memory.
Processor board ID FOX1343GJGC
20 Gigabit Ethernet interfaces
6 Ten Gigabit Ethernet interfaces
32768K bytes of non-volatile configuration memory.
8388608K bytes of physical memory.
1925119K bytes of eUSB flash at bootflash:.
78085207K bytes of SATA hard disk at harddisk:Configuration register is 0x2102

Router# show platform
Chassis type: ASR1013

Slot      Type                State                 Insert time (ago)
--------- ------------------- --------------------- -----------------
 2        ASR1000-SIP40        ok                    1d03h
 2/0      SPA-1X10GE-L-V2      ok                    1d03h
 2/1      SPA-1X10GE-L-V2      ok                    1d03h
 2/2      SPA-1X10GE-L-V2      ok                    1d03h
 2/3      SPA-1X10GE-L-V2      ok                    1d03h
 4        ASR1000-2T+20X1GE    ok                    1d03h
 4/0      BUILT-IN-2T+20X1GE   ok                    1d03h
 R0       ASR1000-RP2          ok, active            1d03h
 R1       ASR1000-RP2          ok, standby           1d03h
 F0       ASR1000-ESP100       ok, active            1d03h
 F1       ASR1000-ESP100       ok, standby           1d03h
 P0       ASR1013-PWR-AC       ok                    1d03h
 P1       ASR1013-PWR-AC       ok                    1d03h
 P2       ASR1013-PWR-AC       ok                    1d03h
 P3       ASR1013-PWR-AC       ps, fail              1d03h

Slot      CPLD Version        Firmware Version
--------- ------------------- ----------------------------------------
 2         00200800            15.3(3r)S
 4         00200800            15.2(1r)S
 R0        10021901            15.3(3r)S
 R1        10021901            15.3(3r)S
 F0        12071700            15.3(3r)S
 F1        12071700            15.3(3r)S
```

```
Router# show version r0 installed
Package: Provisioning File, version: n/a, status: active
  File: bootflash:Active_Dir/packages.conf, on: RP0
  Built: n/a, by: n/a

File SHA1 checksum: a624f70f68c60292f4482433f43afd92487a55c4
Package: rpbase, version: 03.12.01.S.154-2.S, status: active
  File: bootflash:Active_Dir/asr1000rp2-rpbase.03.12.01.S.154-2.S.pkg, on: RP0
  Built: 2013-03-25_18.48, by: mcpre

File SHA1 checksum: 3a9675142898cfac350d4e42f0e37bd9f4e48538
Package: rpcontrol, version: 03.12.01.S.154-2.S, status: active
  File: bootflash:Active_Dir/asr1000rp2-rpcontrol.03.12.01.S.154-2.S.pkg, on: RP0/0
  Built: 2013-03-25_18.48, by: mcpre

File SHA1 checksum: 87b11f863f67fdf2610ee0769b929baab4c3efad
<output removed for brevity>


Router# dir bootflash:Active_Dir
Directory of bootflash:/Active_Dir/
20  -rw-    41104112   Aug 3 2013 15:05:40 +05:30
asr1000rp2-elcbase.03.12.01.S.154-2.S.pkg
21  -rw-    50285296   Aug 3 2013 15:05:40 +05:30
asr1000rp2-elcspa.03.12.01.S.154-2.S.pkg
22  -rw-    82514676   Aug 3 2013 15:05:40 +05:30
asr1000rp2-espbase.03.12.01.S.154-2.S.pkg
23  -rw-   101084628   Aug 3 2013 15:05:40 +05:30
asr1000rp2-espx86base.03.12.01.S.154-2.S.pkg
17  -rw-        9059   Aug 3 2013 15:05:40 +05:30
asr1000rp2-packages-adventerprisek9.03.12.01.S.154-2.S.conf
24  -rw-    29012724   Aug 3 2013 15:05:40 +05:30
asr1000rp2-rpaccess.03.12.01.S.154-2.S.pkg
25  -rw-    49898964   Aug 3 2013 15:05:40 +05:30
asr1000rp2-rpbase.03.12.01.S.154-2.S.pkg
26  -rw-    46557940   Aug 3 2013 15:05:40 +05:30
asr1000rp2-rpcontrol.03.12.01.S.154-2.S.pkg
27  -rw-   114612988   Aug 3 2013 15:05:41 +05:30
asr1000rp2-rpios-adventerprisek9.03.12.01.S.154-2.S.pkg
28  -rw-    41954036   Aug 3 2013 15:05:41 +05:30
asr1000rp2-sipbase.03.12.01.S.154-2.S.pkg
29  -rw-    60957428   Aug 3 2013 15:05:41 +05:30
asr1000rp2-sipspa.03.12.01.S.154-2.S.pkg
19  -rw-        9838   Aug 3 2013 15:05:41 +05:30
packages.conf

1940303872 bytes total (503164928 bytes free)



Router# show redundancy states
       my state = 13
-ACTIVE peer state = 8  -STANDBY HOT
           Mode = Duplex
           Unit = Primary
           Unit ID = 48
Redundancy Mode (Operational) = sso
Redundancy Mode (Configured)  = sso
Redundancy State              = sso
Maintenance Mode = Disabled
Manual Swact = enabled
Communications = Up
client count = 108
client_notification_TMR = 30000 milliseconds
```

```
RF debug mask = 0x0

Router# copy running-config startup-config
Destination filename [startup-config]?
Building configuration...
[OK]

Router# mkdir harddisk:Target_Subs
Create directory filename [Target_Subs]?
Created dir harddisk:/Target_Subs



Router# request platform software package expand file
harddisk:Target_Subs/asr1000rp2-adventerprisek9.03.13.00.S.154-3.S-ext.bin to
harddisk:Target_Subs
Verifying parameters
Validating package type
Copying package files
SUCCESS: Finished expanding all-in-one software package.

#
Router# dir harddisk:Target_Subs
Directory of harddisk:/Target_Subs/
3358722  -rw-   569597380   Aug 4 2013 18:45:38 +05:30
asr1000rp2-adventerprisek9.03.13.00.S.154-3.S-ext.bin
7684099  -rw-    37557200   Aug 4 2013 18:46:43 +05:30
asr1000rp2-elcbase.03.13.00.S.154-3.S-ext.pkg
7684100  -rw-    51194832   Aug 4 2013 18:46:43 +05:30
asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg
7684101  -rw-    80657364   Aug 4 2013 18:46:43 +05:30
asr1000rp2-espbase.03.13.00.S.154-3.S-ext.pkg
7684102  -rw-    95446456   Aug 4 2013 18:46:43 +05:30
asr1000rp2-espx86base.03.13.00.S.154-3.S-ext.pkg
7684097  -rw-        9381   Aug 4 2013 18:46:43 +05:30
asr1000rp2-packages-adventerprisek9.03.13.00.S.154-3.S-ext.conf
7684103  -rw-    23350232   Aug 4 2013 18:46:43 +05:30
asr1000rp2-rpaccess.03.13.00.S.154-3.S-ext.pkg
7684104  -rw-    37694900   Aug 4 2013 18:46:44 +05:30
asr1000rp2-rpbase.03.13.00.S.154-3.S-ext.pkg
7684105  -rw-    45536216   Aug 4 2013 18:46:44 +05:30
asr1000rp2-rpcontrol.03.13.00.S.154-3.S-ext.pkg
7684106  -rw-   118754284   Aug 4 2013 18:46:44 +05:30
asr1000rp2-rpios-adventerprisek9.03.13.00.S.154-3.S-ext.pkg
7684107  -rw-    38380500   Aug 4 2013 18:46:44 +05:30
asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg
7684108  -rw-    61760468   Aug 4 2013 18:46:44 +05:30
asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg
7684098  -rw-       10165   Aug 4 2013 18:46:44 +05:30
packages.conf

78704144384 bytes total (9254879232 bytes free)


Router# copy harddisk:Target_Subs/asr1000rp2-espbase.03.13.00.S.154-3.S-ext.pkg bootflash:
Destination filename [Active_Dir/asr1000rp2-espbase.03.13.00.S.154-3.S-ext.pkg]?
Copy in
progress...CCCC
```

```
Router# copy harddisk:Target_Subs/asr1000rp2-espx86base.03.13.00.S.154-3.S-ext.pkg
bootflash:
Destination filename [Active_Dir/asr1000rp2-espx86base.03.13.00.S.154-3.S-ext.pkg]?
Copy in
progress...CCCC
```

```
Router# copy harddisk:Target_Subs/asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg bootflash:
Destination filename [Active_Dir/asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg]?
Copy in
progress...CCCC
```

```
progress...CCCC
```

```
progress...CCCC
```

```
  Found asr1000rp2-espx86base.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-rpaccess.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-rpbase.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-rpcontrol.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-rpios-adventerprisek9.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg
Verifying image file locations
Inspecting image file types
    WARNING: In-service installation of IOSD package
    WARNING: requires software redundancy on target RP
    WARNING: or on-reboot parameter
    WARNING: Automatically setting the on-reboot flag
    WARNING: In-service installation of RP Base package
    WARNING: requires software reboot of target RP
Processing image file constraints
Creating candidate provisioning file
Finished image file verification

--- Starting candidate package set construction ---
Verifying existing software set
Processing candidate provisioning file
Constructing working set for candidate package set
Constructing working set for running package set
Checking command output
Constructing merge of running and candidate packages
Checking if resulting candidate package set would be complete
Finished candidate package set construction

--- Starting compatibility testing ---
Determining whether candidate package set is compatible
Determining whether installation is valid
Determining whether installation is valid ... skipped
Verifying image type compatibility
Checking IPC compatibility for candidate software
Checking candidate package set infrastructure compatibility
Checking infrastructure compatibility with running software
Checking infrastructure compatibility with running software ... skipped
Checking package specific compatibility
Finished compatibility testing

--- Starting list of software package changes ---
Old files list:
  Removed asr1000rp2-elcbase.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-elcspa.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-espbase.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-espx86base.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-rpaccess.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-rpbase.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-rpcontrol.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-rpios-adventerprisek9.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-sipbase.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-sipspa.03.12.01.S.154-2.S.pkg
New files list:
  Added asr1000rp2-elcbase.03.13.00.S.154-3.S-ext.pkg
  Added asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg
  Added asr1000rp2-espbase.03.13.00.S.154-3.S-ext.pkg
  Added asr1000rp2-espx86base.03.13.00.S.154-3.S-ext.pkg
  Added asr1000rp2-rpaccess.03.13.00.S.154-3.S-ext.pkg
  Added asr1000rp2-rpbase.03.13.00.S.154-3.S-ext.pkg
  Added asr1000rp2-rpcontrol.03.13.00.S.154-3.S-ext.pkg
  Added asr1000rp2-rpios-adventerprisek9.03.13.00.S.154-3.S-ext.pkg
  Added asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg
  Added asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg
```

```
Finished list of software package changes

--- Starting commit of software changes ---
Updating provisioning rollback files
Creating pending provisioning file
Committing provisioning file
Finished commit of software changes

SUCCESS: Software provisioned.  New software will load on reboot.



Router# hw-module slot r1 reload
Proceed with reload of module? [confirm]Y

*Aug  4 19:14:01.721 IST: %IOSXE_OIR-6-OFFLINECARD: Card (rp) offline in slot R1
*Aug  4 19:14:01.761 IST: %REDUNDANCY-3-STANDBY_LOST: Standby processor fault
(PEER_NOT_PRESENT)
*Aug  4 19:14:01.761 IST: %REDUNDANCY-3-STANDBY_LOST: Standby processor fault (PEER_DOWN)
*Aug  4 19:14:01.761 IST: %REDUNDANCY-3-STANDBY_LOST: Standby processor fault
(PEER_REDUNDANCY_STATE_CHANGE)
*Aug  4 19:14:03.584 IST: %RF-5-RF_RELOAD: Peer reload. Reason: EHSA standby down
*Aug  4 19:14:03.594 IST: % Redundancy mode change to SSO
*Aug  4 19:17:35.443 IST: %IOSXE_OIR-6-ONLINECARD: Card (rp) online in slot R1
*Aug  4 19:17:48.061 IST: %REDUNDANCY-5-PEER_MONITOR_EVENT: Active detected a standby
insertion (raw-event=PEER_FOUND(4))
*Aug  4 19:17:48.061 IST: %REDUNDANCY-5-PEER_MONITOR_EVENT: Active detected a standby
insertion (raw-event=PEER_REDUNDANCY_STATE_CHANGE(5))
*Aug  4 19:17:51.003 IST: %REDUNDANCY-3-IPC: IOS versions do not match.
*Aug  4 19:19:08.380 IST: %NBAR_HA-5-NBAR_INFO: NBAR sync DONE!
*Aug  4 19:19:08.797 IST: %HA_CONFIG_SYNC-6-BULK_CFGSYNC_SUCCEED: Bulk Sync succeeded
*Aug  4 19:19:08.798 IST: %RF-5-RF_TERMINAL_STATE: Terminal state reached for (SSO)

Router# request platform software package install rp 0 file
bootflash:Active_Dir/asr1000rp2-{sipbase,sipspa}*03.13.00.S.154-3.S-ext*.pkg slot 2 force
--- Starting local lock acquisition on R0 ---
Finished local lock acquisition on R0

--- Starting installation state synchronization ---
Finished installation state synchronization

--- Starting file path checking ---
Finished file path checking

--- Starting image file verification ---
Checking image file names
Locating image files and validating name syntax
  Found asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg
Verifying image file locations
Inspecting image file types
Processing image file constraints
Creating candidate provisioning file
Finished image file verification

--- Starting candidate package set construction ---
Verifying existing software set
Processing candidate provisioning file
Constructing working set for candidate package set
Constructing working set for running package set
Checking command output
Constructing merge of running and candidate packages
Checking if resulting candidate package set would be complete
Finished candidate package set construction
```

```
--- Starting compatibility testing ---
Determining whether candidate package set is compatible

WARNING: Candidate software combination not found in compatibility database
WARNING: Determining whether installation is valid
Creating matrix_file by locate_latest_matrix_file  /tmp/issu/provision/sw

WARNING: Candidate software combination not found in compatibility database
WARNING: Candidate software combination not found in compatibility database
WARNING: Software sets are identified as compatible
Verifying image type compatibility
Checking IPC compatibility with running software
Checking candidate package set infrastructure compatibility
Checking infrastructure compatibility with running software
Checking package specific compatibility
Finished compatibility testing

--- Starting impact testing ---
Checking operational impact of change
Finished impact testing

--- Starting list of software package changes ---
No old package files removed
New files list:
  Added asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg
  Added asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg
Finished list of software package changes

--- Starting commit of software changes ---
Updating provisioning rollback files
Creating pending provisioning file
Committing provisioning file
Finished commit of software changes

--- Starting analysis of software changes ---
Finished analysis of software changes

--- Starting update running software ---
Blocking peer synchronization of operating information
Creating the command set placeholder directory
  Finding latest command set
  Finding latest command shortlist lookup file
  Finding latest command shortlist file
  Assembling CLI output libraries
  Assembling CLI input libraries
  Assembling Dynamic configuration files
  Applying interim IPC and database definitions
  Replacing running software
  Replacing CLI software
  Restarting software
  Generating software version information
  Notifying running software of updates
  Unblocking peer synchronization of operating information
Unmounting old packages
Cleaning temporary installation files
  Finished update running software

SUCCESS: Finished installing software.

*Aug  4 19:21:45.424 IST:
%IOSXE_OIR-6-ONLINECARD: Card (cc) online in slot 2
*Aug  4 19:21:48.382 IST: %IOSXE_OIR-6-INSSPA: SPA inserted in subslot 2/0
*Aug  4 19:21:48.733 IST: %IOSXE_OIR-6-INSSPA: SPA inserted in subslot 2/1
```

```
*Aug  4 19:21:49.083 IST: %IOSXE_OIR-6-INSSPA: SPA inserted in subslot 2/2
*Aug  4 19:21:49.430 IST: %IOSXE_OIR-6-INSSPA: SPA inserted in subslot 2/3
*Aug  4 19:21:58.121 IST: %LINK-3-UPDOWN: SIP2/0: Interface EOBC2/1, changed state to up
*Aug  4 19:22:02.302 IST: %SPA_OIR-6-ONLINECARD: SPA (SPA-1X10GE-L-V2) online in subslot
2/0
*Aug  4 19:22:02.282 IST: %TRANSCEIVER-6-INSERTED: SIP2/0: transceiver module inserted in
TenGigabitEthernet2/0/0
*Aug  4 19:22:02.518 IST: %LINK-3-UPDOWN: SIP2/1: Interface EOBC2/1, changed state to up
*Aug  4 19:22:06.113 IST: %SPA_OIR-6-ONLINECARD: SPA (SPA-1X10GE-L-V2) online in subslot
2/1
*Aug  4 19:22:06.082 IST: %TRANSCEIVER-6-INSERTED: SIP2/1: transceiver module inserted in
TenGigabitEthernet2/1/0
*Aug  4 19:22:08.080 IST: %LINK-3-UPDOWN: SIP2/2: Interface EOBC2/1, changed state to up
*Aug  4 19:22:11.627 IST: %SPA_OIR-6-ONLINECARD: SPA (SPA-1X10GE-L-V2) online in subslot
2/2
*Aug  4 19:22:11.616 IST: %TRANSCEIVER-6-INSERTED: SIP2/2: transceiver module inserted in
TenGigabitEthernet2/2/0
*Aug  4 19:22:12.523 IST: %LINK-3-UPDOWN: SIP2/3: Interface EOBC2/1, changed state to up
*Aug  4 19:22:16.657 IST: %SPA_OIR-6-ONLINECARD: SPA (SPA-1X10GE-L-V2) online in subslot
2/3
*Aug  4 19:22:16.648 IST: %TRANSCEIVER-6-INSERTED: SIP2/3: transceiver module inserted in
TenGigabitEthernet2/3/0


Router# issu commitversion
--- Starting local lock acquisition on R0 ---
Finished local lock acquisition on R0

--- Starting installation changes ---
Cancelling rollback timer
Finished installation changes

SUCCESS: Installation changes committed


Router# request platform software package install rp 0 file
bootflash:Active_Dir/asr1000rp2-{elcbase,elcspa}*03.13.00.S.154-3.S-ext*.pkg slot 4
--- Starting local lock acquisition on R0 ---
Finished local lock acquisition on R0

--- Starting installation state synchronization ---
Finished installation state synchronization

--- Starting file path checking ---
Finished file path checking

--- Starting image file verification ---
Checking image file names
Locating image files and validating name syntax
  Found asr1000rp2-elcbase.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg
Verifying image file locations
Inspecting image file types
Processing image file constraints
Creating candidate provisioning file
Finished image file verification

--- Starting candidate package set construction ---
Verifying existing software set
Processing candidate provisioning file
Constructing working set for candidate package set
Constructing working set for running package set
Checking command output Constructing merge of running and candidate packages
Checking if resulting candidate package set would be complete
```

```
Finished candidate package set construction

--- Starting compatibility testing ---
Determining whether candidate package set is compatible

WARNING:Candidate software combination not found in compatibility database
WARNING:Determining whether installation is valid
Creating matrix_file by locate_latest_matrix_file  /tmp/issu/provision/sw

WARNING:Candidate software combination not found in compatibility database
WARNING:Candidate software combination not found in compatibility database
WARNING:Software sets are identified as compatible
Verifying image type compatibility
Checking IPC compatibility with running software
Checking candidate package set infrastructure compatibility
Checking infrastructure compatibility with running software
Checking package specific compatibility
Finished compatibility testing

--- Starting impact testing ---
Checking operational impact of change
Finished impact testing

--- Starting list of software package changes ---
No old package files removed
New files list:
  Added asr1000rp2-elcbase.03.13.00.S.154-3.S-ext.pkg
  Added asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg
Finished list of software package changes

--- Starting commit of software changes ---
Updating provisioning rollback files
Creating pending provisioning file
Committing provisioning file
Finished commit of software changes

--- Starting analysis of software changes ---
Finished analysis of software changes

--- Starting update running software ---
Blocking peer synchronization of operating information
Creating the command set placeholder directory
  Finding latest command set
  Finding latest command shortlist lookup file
  Finding latest command shortlist file
  Assembling CLI output libraries
  Assembling CLI input libraries
  Assembling Dynamic configuration files
  Applying interim IPC and database definitions
  Replacing running software
  Replacing CLI software
  Restarting software
  Applying final IPC and database definitions
  Generating software version information
  Notifying running software of updates
  Unblocking peer synchronization of operating information
Unmounting old packages
Cleaning temporary installation files
  Finished update running software

SUCCESS: Finished installing software.


Router# issu commitversion
```

```
Starting local lock acquisition on R0 ---
Finished local lock acquisition on R0

--- Starting installation changes ---
Cancelling rollback timer
Finished installation changes

SUCCESS: Installation changes committed


Router# request platform software package install rp 0 file
bootflash:Active_Dir/asr1000rp2-esp*03.13.00.S.154-3.S-ext*.pkg slot 1
--- Starting local lock acquisition on R0 ---
Finished local lock acquisition on R0

--- Starting installation state synchronizatio---
Finished installation state synchronization

--- Starting file path checking ---
Finished file path checking

--- Starting image file verification ---
Checking image file names
Locating image files and validating name syntax
  Found asr1000rp2-espbase.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-espx86base.03.13.00.S.154-3.S-ext.pkg
Verifying image file locations
Inspecting image file types
Processing image file constraints
Creating candidate provisioning file
Finished image file verification

--- Starting candidate package set construction ---
Verifying existing software set
Processing candidate provisioning file
Constructing working set for candidate package set
Constructing working set for running package set
Checking command output
Constructing merge of running and candidate packages
Checking if resulting candidate package set would be complete
Finished candidate package set construction

--- Starting compatibility testing ---
Determining whether candidate package set is compatible

WARNING:Candidate software combination not found in compatibility database
WARNING:Determining whether installation is valid
Creating matrix_file by locate_latest_matrix_file  /tmp/issu/provision/sw

WARNING:Candidate software combination not found in compatibility database
WARNING:Candidate software combination not found in compatibility database
WARNING:Software sets are identified as compatible
Verifying image type compatibility
Checking IPC compatibility with running software
Checking candidate package set infrastructure compatibility
Checking infrastructure compatibility with running software
Checking package specific compatibility
Finished compatibility testing

--- Starting impact testing ---
Checking operational impact of change
Finished impact testing
```

```
--- Starting list of software package changes ---
No old package files removed
New files list:
  Added asr1000rp2-espbase.03.13.00.S.154-3.S-ext.pkg
  Added asr1000rp2-espx86base.03.13.00.S.154-3.S-ext.pkg
Finished list of software package changes

--- Starting commit of software changes ---
Updating provisioning rollback files
Creating pending provisioning file
Committing provisioning file
Finished commit of software changes

--- Starting analysis of software changes ---
Finished analysis of software changes

--- Starting update running software ---
Blocking peer synchronization of operating information
Creating the command set placeholder directory
  Finding latest command set
  Finding latest command shortlist lookup file
  Finding latest command shortlist file
  Assembling CLI output libraries
  Assembling CLI input libraries
  Assembling Dynamic configuration files
  Applying interim IPC and database definitions
  Replacing running software
  Replacing CLI software
  Restarting software
    Restarting ESP1
  Applying final IPC and database definitions

*Aug  4 19:29:16.751 IST: %IOSXE_OIR-6-OFFLINECARD: Card (fp) offline in slot F1
*Aug  4 19:29:18.172 IST: %CMRP-6-FP_HA_STATUS: R0/0: cmand:  F0 redundancy state is
Active with no Standby
Generating software version information
Notifying running software of updates
  Unblocking peer synchronization of operating information
Unmounting old packages
Cleaning temporary installation files
  Finished update running software

SUCCESS: Finished installing software.


*Aug  4 19:30:50.972 IST: %CPPHA-7-START: F1: cpp_ha:  CPP 0 preparing image
/tmp/sw/fp/1/0/fpx86/mount/usr/cpp/bin/qfp-ucode-esp40
*Aug  4 19:30:51.362 IST: %CPPHA-7-START: F1: cpp_ha:  CPP 0 startup init image
/tmp/sw/fp/1/0/fpx86/mount/usr/cpp/bin/qfp-ucode-esp40
*Aug  4 19:30:53.088 IST: %IOSXE_OIR-6-ONLINECARD: Card (fp) online in slot F1


Router# issu commitversion
--- Starting local lock acquisition on R0 ---
Finished local lock acquisition on R0

--- Starting installation changes ---
Cancelling rollback timer
Finished installation changes

SUCCESS: Installation changes committed
```

```
Router# request platform software package install rp 0 file
bootflash:Active_Dir/asr1000rp2-esp*03.13.00.S.154-3.S-ext*.pkg slot 0
--- Starting local lock acquisition on R0 ---
Finished local lock acquisition on R0

--- Starting installation state synchronization ---
Finished installation state synchronization

--- Starting file path checking ---
Finished file path checking

--- Starting image file verification ---
Checking image file names
Locating image files and validating name syntax
  Found asr1000rp2-espbase.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-espx86base.03.13.00.S.154-3.S-ext.pkg
Verifying image file locations
Inspecting image file types
Processing image file constraints
Creating candidate provisioning file

*Aug  4 19:31:14.730 IST: %CPPHA-7-START: F1: cpp_ha:  CPP 0 running init image
/tmp/sw/fp/1/0/fpx86/mount/usr/cpp/bin/qfp-ucode-esp40
*Aug  4 19:31:15.079 IST: %CPPHA-7-READY: F1: cpp_ha:  CPP 0 loading and initialization
completeFinished image file verification

--- Starting candidate package set construction ---
Verifying existing software set

*Aug  4 19:31:15.309 IST: %IOSXE-6-PLATFORM: F1: cpp_cp: Process
CPP_PFILTER_EA_EVENT__API_CALL__REGISTERProcessing candidate provisioning file
*Aug  4 19:31:18.010 IST: %CMRP-6-FP_HA_STATUS: R0/0: cmand:  F0 redundancy state is
Active with ready StandbyConstructing working set for candidate package set
Constructing working set for running package set
Checking command output
Constructing merge of running and candidate packages
Checking if resulting candidate package set would be complete
Finished candidate package set construction

--- Starting compatibility testing ---
Determining whether candidate package set is compatible

WARNING:Candidate software combination not found in compatibility database
WARNING:Determining whether installation is valid
Creating matrix_file by locate_latest_matrix_file /tmp/issu/provision/sw
Software sets are identified as compatible
Verifying image type compatibility
Checking IPC compatibility with running software
Checking candidate package set infrastructure compatibility
Checking infrastructure compatibility with running software
Checking package specific compatibility
Finished compatibility testing

--- Starting impact testing ---
Checking operational impact of change
Finished impact testing

--- Starting list of software package changes ---
Old files list:
  Removed asr1000rp2-espbase.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-espx86base.03.12.01.S.154-2.S.pkg
No new package files added
Finished list of software package changes
```

```
--- Starting commit of software changes ---
Updating provisioning rollback files
Creating pending provisioning file
Committing provisioning file
Finished commit of software changes

--- Starting analysis of software changes ---
Finished analysis of software changes

--- Starting update running software ---
Blocking peer synchronization of operating information
Creating the command set placeholder directory
  Finding latest command set
  Finding latest command shortlist lookup file
  Finding latest command shortlist file
  Assembling CLI output libraries
  Assembling CLI input libraries
  Assembling Dynamic configuration files
  Applying interim IPC and database definitions
  Replacing running software
  Replacing CLI software
  Restarting software
  Restarting ESP0
  Applying final IPC and database definitions

*Aug  4 19:32:46.187 IST: %IOSXE_OIR-6-OFFLINECARD: Card (fp) offline in slot F0
*Aug  4 19:32:46.539 IST: %CMRP-6-FP_HA_STATUS: R0/0: cmand:  F1 redundancy state is
Active
Generating software version information
  Notifying running software of updates
  Unblocking peer synchronization of operating information
Unmounting old packages
Cleaning temporary installation files
  Finished update running software

SUCCESS: Finished installing software.


*Aug  4 19:34:19.748 IST: %CPPHA-7-START: F0: cpp_ha:  CPP 0 preparing image
/tmp/sw/fp/0/0/fpx86/mount/usr/cpp/bin/qfp-ucode-esp40
*Aug  4 19:34:20.139 IST: %CPPHA-7-START: F0: cpp_ha:  CPP 0 startup init image
/tmp/sw/fp/0/0/fpx86/mount/usr/cpp/bin/qfp-ucode-esp40
*Aug  4 19:34:21.858 IST: %IOSXE_OIR-6-ONLINECARD: Card (fp) online in slot F0
*Aug  4 19:34:43.609 IST: %CPPHA-7-START: F0: cpp_ha:  CPP 0 running init image
/tmp/sw/fp/0/0/fpx86/mount/usr/cpp/bin/qfp-ucode-esp40
*Aug  4 19:34:43.958 IST: %CPPHA-7-READY: F0: cpp_ha:  CPP 0 loading and initialization
complete
*Aug  4 19:34:44.190 IST: %IOSXE-6-PLATFORM: F0: cpp_cp: Process
CPP_PFILTER_EA_EVENT__API_CALL__REGISTER
*Aug  4 19:34:46.890 IST: %CMRP-6-FP_HA_STATUS: R0/0: cmand:  F0 redundancy state is
Standby

Router# show platform
Chassis type: ASR1013

Slot      Type                State                 Insert time (ago)
--------- ------------------- --------------------- -----------------
 2         ASR1000-SIP40       ok                    1d04h
 2/0       SPA-1X10GE-L-V2     ok                    1d04h
 2/1       SPA-1X10GE-L-V2     ok                    1d04h
 2/2       SPA-1X10GE-L-V2     ok                    1d04h
 2/3       SPA-1X10GE-L-V2     ok                    1d04h
 4         ASR1000-2T+20X1GE   ok                    1d04h
 4/0       BUILT-IN-2T+20X1GE  ok                    1d04h
```

```
 R0        ASR1000-RP2         ok, active            1d04h
 R1        ASR1000-RP2         ok, standby           1d04h
 F0        ASR1000-ESP100      ok, standby           1d04h
 F1        ASR1000-ESP100      ok, active            1d04h
 P0        ASR1013-PWR-AC      ok                    1d04h
 P1        ASR1013-PWR-AC      ok                    1d04h
 P2        ASR1013-PWR-AC      ok                    1d04h
 P3        ASR1013-PWR-AC      ps, fail              1d04h

Slot      CPLD Version        Firmware Version
---------------------------------------------------------------------------
 2         00200800            15.3(3r)S
 4         00200800            15.2(1r)S
 R0        10021901            15.3(3r)S
 R1        10021901            15.3(3r)S
 F0        12071700            15.3(3r)S
 F1        12071700            15.3(3r)S


Router# issu commitversion

Starting local lock acquisition on R0 ---
Finished local lock acquisition on R0

--- Starting installation changes ---
Cancelling rollback timer
Finished installation changes

SUCCESS: Installation changes committed


Router# request platform software package install rp 0 file
bootflash:Active_Dir/asr1000rp2*03.13.00.S.154-3.S-ext*.pkg force
--- Starting local lock acquisition on R0 ---
Finished local lock acquisition on R0

--- Starting installation state synchronization ---
Finished installation state synchronization

--- Starting file path checking ---
Finished file path checking

--- Starting image file verification ---
Checking image file names
Locating image files and validating name syntax
  Found asr1000rp2-elcbase.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-espbase.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-espx86base.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-rpaccess.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-rpbase.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-rpcontrol.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-rpios-adventerprisek9.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg
Verifying image file locations
Inspecting image file types

WARNING: In-service installation of IOSD package
    WARNING: requires software redundancy on target RP
    WARNING: or on-reboot parameter
    WARNING: Automatically setting the on-reboot flag
    WARNING: In-service installation of RP Base package
    WARNING: requires software reboot of target RP
```

```
Processing image file constraints
Creating candidate provisioning file
Finished image file verification

--- Starting candidate package set construction ---
Verifying existing software set
Processing candidate provisioning file
Constructing working set for candidate package set
Constructing working set for running package set
Checking command output
Constructing merge of running and candidate packages
Checking if resulting candidate package set would be complete
Finished candidate package set construction

--- Starting compatibility testing ---
Determining whether candidate package set is compatible
Determining whether installation is valid
Determining whether installation is valid ... skipped
Verifying image type compatibility
Checking IPC compatibility for candidate software
Checking candidate package set infrastructure compatibility
Checking infrastructure compatibility with running software
Checking infrastructure compatibility with running software ... skipped
Checking package specific compatibility
Finished compatibility testing

--- Starting list of software package changes ---
Old files list:
  Removed asr1000rp2-elcbase.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-elcspa.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-rpaccess.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-rpbase.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-rpcontrol.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-rpios-adventerprisek9.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-sipbase.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-sipspa.03.12.01.S.154-2.S.pkg
New files list:
  Added asr1000rp2-rpaccess.03.13.00.S.154-3.S-ext.pkg
  Added asr1000rp2-rpbase.03.13.00.S.154-3.S-ext.pkg
  Added asr1000rp2-rpcontrol.03.13.00.S.154-3.S-ext.pkg
  Added asr1000rp2-rpios-adventerprisek9.03.13.00.S.154-3.S-ext.pkg
Finished list of software package changes

--- Starting commit of software changes ---
Updating provisioning rollback files
Creating pending provisioning file
Committing provisioning file
Finished commit of software changes

SUCCESS: Software provisioned.  New


software will load on reboot.


Router# show version R0 provisioned
Package: Provisioning File, version: n/a, status: active
  File: bootflash:Active_Dir/packages.conf, on: RP0
  Built: n/a, by: n/a
  File SHA1 checksum: c79075780592aec1312725f4a2357a034fda2d3b

Package: rpbase, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-rpbase.03.13.00.S.154-3.S-ext.pkg, on: RP0
  Built: 2013-07-25_22.55, by: mcpre
```

```
  File SHA1 checksum: 4f655c54bb95b4dfa24a0d25ebf97cf8527c69e9

Package: rpcontrol, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-rpcontrol.03.13.00.S.154-3.S-ext.pkg, on: RP0/0
  Built: 2013-07-25_22.55, by: mcpre
  File SHA1 checksum: 8a0a45ea5c7a656c0eef6726174461584f182c78

Package: rpios-adventerprisek9, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-rpios-adventerprisek9.03.13.00.S.154-3.S-ext.pkg,
on: RP0/0Built: 2013-07-25_23.00, by: mcpre
  File SHA1 checksum: 85e9eab826bff2194ef568a56c76453625383ad2

   File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipbase, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg, on: SIP2
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 3b6a4838972840a995ff22e73fd2bae910b268a7

<some output removed for brevity>



Router# show version R0 provisioned
Package: Provisioning File, version: n/a, status: active
  File: bootflash:Active_Dir/packages.conf, on: RP0
  Built: n/a, by: n/a
  File SHA1 checksum: c79075780592aec1312725f4a2357a034fda2d3b

Package: rpbase, version: 03.12.01.S.154-2.S, status: active
  File: bootflash:Active_Dir/asr1000rp2-rpbase.03.12.01.S.154-2.S.pkg, on: RP0
  Built: 2013-03-25_18.48, by: mcpre
  File SHA1 checksum: 3a9675142898cfac350d4e42f0e37bd9f4e48538

Package: rpcontrol, version: 03.12.01.S.154-2.S, status: active
  File: bootflash:Active_Dir/asr1000rp2-rpcontrol.03.12.01.S.154-2.S.pkg, on: RP0/0
  Built: 2013-03-25_18.48, by: mcpre
  File SHA1 checksum: 87b11f863f67fdf2610ee0769b929baab4c3efad

Package: rpios-adventerprisek9, version: 03.12.01.S.154-2.S, status: active
  File: bootflash:Active_Dir/asr1000rp2-rpios-adventerprisek9.03.12.01.S.154-2.S.pkg, on:
RP0/0   Built: 2013-03-25_18.51, by: mcpre
  File SHA1 checksum: b487136319da0a327844d353c77e533c53c56053

Package: rpaccess, version: 03.12.01.S.154-2.S, status: active
  File: bootflash:Active_Dir/asr1000rp2-rpaccess.03.12.01.S.154-2.S.pkg, on: RP0/0
  Built: 2013-03-25_18.48, by: mcpre
  File SHA1 checksum: 032bea36f74b19977b363243c99f02413b54104d

<some output removed for brevity>


Router# redundancy force-switchover
Proceed with switchover to standby RP? [confirm]
<output removed for brevity>


Router# request platform software package clean
Cleaning up unnecessary package files
No path specified, will use booted path bootflash:Active_Dir/packages.conf
Cleaning bootflash:Active_Dir
  Scanning boot directory for packages ... done.
  Preparing packages list to delete ... asr1000rp2-elcbase.03.13.00.S.154-3.S-ext.pkg
      File is in use, will not delete..asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg
```

```
      File is in use, will not delete...asr1000rp2-espbase.03.13.00.S.154-3.S-ext.pkg
      File is in use, will not delete..asr1000rp2-espx86base.03.13.00.S.154-3.S-ext.pkg
      File is in use, will not delete...asr1000rp2-rpaccess.03.13.00.S.154-3.S-ext.pkg
      File is in use, will not delete... asr1000rp2-rpbase.03.13.00.S.154-3.S-ext.pkg
      File is in use, will not delete.
    asr1000rp2-rpcontrol.03.13.00.S.154-3.S-ext.pkg
      File is in use, will not
delete...asr1000rp2-rpios-adventerprisek9.03.13.00.S.154-3.S-ext.pkg
      File is in use, will not delete...asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg
      File is in use, will not delete...   asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg
      File is in use, will not delete...packages.conf
      File is in use, will not delete...  done.

Files that will be deleted: asr1000rp2-elcbase.03.12.01.S.154-2.S.pkg
  asr1000rp2-elcspa.03.12.01.S.154-2.S.pkg
  asr1000rp2-espbase.03.12.01.S.154-2.S.pkg
  asr1000rp2-espx86base.03.12.01.S.154-2.S.pkg
  asr1000rp2-packages-adventerprisek9.03.12.01.S.154-2.S.conf
  asr1000rp2-rpaccess.03.12.01.S.154-2.S.pkg
  asr1000rp2-rpbase.03.12.01.S.154-2.S.pkg
  asr1000rp2-rpcontrol.03.12.01.S.154-2.S.pkg
  asr1000rp2-rpios-adventerprisek9.03.12.01.S.154-2.S.pkg
  asr1000rp2-sipbase.03.12.01.S.154-2.S.pkg
  asr1000rp2-sipspa.03.12.01.S.154-2.S.pkg
  packages.conf.00-

Do you want to proceed? [confirm]y
  Deleting file bootflash:Active_Dir/asr1000rp2-elcbase.03.12.01.S.154-2.S.pkg ... done.
  Deleting file bootflash:Active_Dir/asr1000rp2-elcspa.03.12.01.S.154-2.S.pkg ... done.
  Deleting file bootflash:Active_Dir/asr1000rp2-espbase.03.12.01.S.154-2.S.pkg ... done.
  Deleting file bootflash:Active_Dir/asr1000rp2-espx86base.03.12.01.S.154-2.S.pkg ...
done.
  Deleting file
bootflash:Active_Dir/asr1000rp2-packages-adventerprisek9.03.12.01.S.154-2.S.conf ... done.
  Deleting file bootflash:Active_Dir/asr1000rp2-rpaccess.03.12.01.S.154-2.S.pkg ... done.
  Deleting file bootflash:Active_Dir/asr1000rp2-rpbase.03.12.01.S.154-2.S.pkg ... done.
  Deleting file bootflash:Active_Dir/asr1000rp2-rpcontrol.03.12.01.S.154-2.S.pkg ... done.
  Deleting file
bootflash:Active_Dir/asr1000rp2-rpios-adventerprisek9.03.12.01.S.154-2.S.pkg ... done.
  Deleting file bootflash:Active_Dir/asr1000rp2-sipbase.03.12.01.S.154-2.S.pkg ... done.
  Deleting file bootflash:Active_Dir/asr1000rp2-sipspa.03.12.01.S.154-2.S.pkg ... done.
  Deleting file bootflash:Active_Dir/packages.conf.00- ... done.

SUCCESS: Files deleted...
```

In Service One-Shot Software Upgrade Procedure

In service one-shot software upgrade procedure is an alternate to the standard ISSU procedure (see, ISSU Upgrade Procedures, page 5-6) that enables you to upgrade or downgrade software using a single command. One-shot ISSU needs minimal user intervention or monitoring. Unlike the standard ISSU procedures, once the upgrade is initiated, the upgrade process cannot be cancelled.

The one-shot upgrade procedure is divided into stages. When a failure occurs, the command execution is stalled and users have to perform the rollback tasks manually. Necessary switchovers are automatically taken care of in one of the upgrade stages. During a switchover, the console and its output are lost. Additional commands are used to connect back to the console.

Note One-shot upgrade does not support multiple upgrades at the same time.

The **request platform software package install node file** *consolidated file name* **interface-module-delay** *secs* command is used for the one-shot ISSU procedure.

interface-module-delay is an optional parameter.

The **interface-module-delay** option specifies the time in seconds to wait after the first SIP and SPA combination has completed its upgrade or reboot before starting the upgrade of the next SIP and SPA. A minimum value of 60 seconds is recommended to ensure that all of the previously restarted SPAs are operational.

Example:

```
router# request platform software package install node file bootflash
interface-module-delay 60
```

The SIP-delay option enables you to delay the SIP restart duration thus enabling you to choose the restart time to minimize traffic loss.

The **request platform software package install node attach** command enables the users to view the last run log that contains the reports of all the stages of the one-shot upgrade.

The **request platform software package install file mdr** command enables the users to initiate the ISSU procedure using MDR. For more information see, "Minimal Disruptive Restart ISSU" section on page 5-152.

ISSU Procedures (Prior to Cisco IOS XE Release 2.1.2)

These instructions should be followed only if you are upgrading using ISSU to a pre-Cisco IOS XE 2.1.2 release. If you are using Cisco IOS XE Release 2.2.1 or later, follow the earlier instructions in this chapter to perform your ISSU upgrade.

This section contains the following topics:

- Using ISSU to Perform a Consolidated Package Upgrade in a Dual Route Processor Configuration (Prior to Cisco IOS XE 2.1.2), page 5-72
- Using ISSU to Upgrade Subpackages (Prior to Cisco IOS XE Release 2.1.2), page 5-72

Using ISSU to Perform a Consolidated Package Upgrade in a Dual Route Processor Configuration (Prior to Cisco IOS XE 2.1.2)

This procedure is identical to the procedure used to upgrade post-Cisco IOS XE 2.2 images using ISSU.

See the "Using ISSU to Perform a Consolidated Package Upgrade in a Dual Route Processor Configuration" section on page 5-7 for instructions on performing this process.

Using ISSU to Upgrade Subpackages (Prior to Cisco IOS XE Release 2.1.2)

This procedure will work on single RPs configured to run two IOS processes or on routers configured with dual RPs.

Versions of Cisco IOS XE Release 2.1 prior to Cisco IOS XE Release 2.1.2 and versions of Cisco IOS XE Release 2.2 prior to Cisco IOS XE Release 2.2.1 and installations not booted from the RPBase subpackage on Cisco IOS XE Release 2.2.1 or 2.1.2 require a different ISSU upgrade procedure.

Step 1 Copy all subpackages other than the **RPBase** subpackage into the same directory.

Step 2 Install the RP subpackages from the directory simultaneously by using a wildcard statement to capture all of the RP subpackages. This command should capture the RPControl, RPAccess, and RPIOS subpackages without capturing the RPBase subpackage and should be done using the standby bay.

Use the asr1000rp1-rp**version*.pkg syntax instead of using the {*pattern*} syntax to complete this upgrade:

request platform software package install rp 1 file stby-bootflash:asr1000rp*02.02.01.122-33.XNB1*.pkg bay 1 force

Step 3 Wait for the system to reach SSO ready state. The **show platform** command can be used to monitor whether both IOS instances are active, and a system message will indicate SSO state has been reached.

Step 4 Use the **redundancy force-switchover** command to trigger an IOS switchover.

Step 5 Wait for the system to reach SSO ready state. Like in step 3, the state can be monitored by checking system messages or by entering the **show platform** command.

Step 6 Install the RP subpackages from the directory simultaneously by using a wildcard statement to capture all of the RP subpackages. This command should capture the RPControl, RPAccess, and RPIOS subpackages without capturing the RPBase subpackage and should be done using the standby bay.

Use the "asr1000rp1-rp**version*.pkg syntax instead of using the {*pattern*} syntax to complete this upgrade:

```
request platform software package install rp 0 file
stby-bootflash:asr1000rp*02.02.01.122-33.XNB1*.pkg bay 0 force
```

Step 7 Install the SIPBASE and SIPSPA packages on each SIP:

```
Router# request platform software package install rp 0 file
bootflash:asr1000rp1-sip*02.02.01.122-33.XNB1*.pkg slot 4 force
Router# request platform software package install rp 0 file
bootflash:asr1000rp1-sip*02.02.01.122-33.XNB1*.pkg slot 1 force
Router# request platform software package install rp 0 file
bootflash:asr1000rp1-sip*02.02.01.122-33.XNB1*.pkg slot 2 force
```

Step 8 Install the ESPBASE package on the ESP:

```
Router# request platform software package install rp 0 file
bootflash:asr1000rp1-esp*02.02.01.122-33.XNB1*.pkg
```

Step 9 Copy the RPBase subpackage into the directory, then upgrade all of the RP subpackages simultaneously. This step will upgrade the RPBase subpackage, which is the last remaining subpackage that requires an upgrade:

```
request platform software package install rp 1 file
bootflash:asr1000rp*02.02.01.122-33.XNB1*.pkg force
```

Step 10 Enter **show version provisioned** to confirm that all of the software has been updated.

Step 11 Reload the router using the **reload** command when appropriate to complete the installation.

Downgrades in versions that do not support the {*pattern*} syntax are problematic because an RPBASE package for the downgrade version will always be present if an upgrade was previously performed. Removing and restaging all packages will work in this case, but they effectively put the system into a state where if the RP reloads prior to the installation being completed, the RP may no longer be bootable.

Upgrade Process with Service Impact for Nonredundant Platforms

Subpackage software upgrade is supported for nonredundant platforms such as Cisco ASR 1001 Router, Cisco ASR1001-X, Cisco ASR 1002, Cisco ASR 1002-X, and ASR 1004 Routers in subpackage mode. This is because the software upgrade procedure on these chassis types requires an RP reload when upgrading the RPBase subpackage at the last step.

During the software upgrade process, there will be outage on the control plane as the entire platform is rebooted so that access to the router operating system and ROMmon is lost for a period of time.

For non-hardware-redundant chassis types, SIP impact can be mitigated by installing SIPs one slot at a time if SPAs are redundant across SIPs (such as when using Gigabit Etherchannel). ESP redundancy provides similar capability for the ESP allowing hitless upgrade of a chassis from one software release to another. Consolidated package mode does not provide such a per-slot staging option and always incurs a traffic loss equivalent to simultaneous OIR of all SIPs.

Note The Cisco ASR 1002 and ASR 1002-F Routers come by default with 4-GB DRAM. The Cisco ASR 1001 Router comes by default with 4-GB DRAM, and is upgradeable up to 8-GB or 16-GB DRAM.

This section explains how to upgrade subpackages on a Cisco ASR 1001, Cisco ASR 1001-X, Cisco ASR1002, Cisco ASR 1002-X, or Cisco ASR 1004 Router. It contains the following sections:

- Configuring SSO on a Cisco ASR 1001, Cisco ASR 1001-X, Cisco ASR 1002, Cisco ASR 1002-X, or Cisco ASR 1004 Router, page 5-74
- Using Subpackages for Software Upgrade on a Cisco ASR 1001 Router, Cisco ASR 1001-X Router or a Cisco ASR 1002-X Router, page 5-76
- Using Subpackages for Software Upgrade on a Cisco ASR 1002 Router or Cisco ASR 1004 Router (software upgrade Command Set), page 5-91
- Using Subpackages for Software Upgrade on a Cisco ASR 1002 Router or Cisco ASR 1004 Router (request platform Command Set), page 5-121

Configuring SSO on a Cisco ASR 1001, Cisco ASR 1001-X, Cisco ASR 1002, Cisco ASR 1002-X, or Cisco ASR 1004 Router

The following instructions show how to configure SSO on a Cisco ASR 1001, Cisco ASR 1001-X, Cisco ASR 1002, Cisco ASR 1002-X, and Cisco ASR 1004 Routers. The standby IOS process is created automatically as part of these configuration steps.

Step 1 (Optional) Enter the **show version** command to confirm the amount of DRAM on your router:

```
Router# show version
<some output removed for brevity>
32768K bytes of non-volatile configuration memory.
4194304K bytes of physical memory.
921599K bytes of eUSB flash at bootflash:.
39004543K bytes of SATA hard disk at bootflash:.

Configuration register is 0x2102
```

In the example **show version** output, the router has 4 GB of DRAM memory.

If you are using a Cisco ASR 1001, Cisco ASR 1001-X, Cisco ASR 1002, Cisco ASR 1002-X, or Cisco ASR 1004 Router with less than required DRAM memory on the RP, SSO cannot be configured on your RP. For more information, see Prerequisites for Software Upgrade Processes, page 5-2.

Step 2 (Optional) Enter **show redundancy states** to see the current HA configuration:

```
Router# show redundancy states
       my state = 13 -ACTIVE
     peer state = 1  -DISABLED
           Mode = Simplex
        Unit ID = 6

Redundancy Mode (Operational) = Non-redundant
Redundancy Mode (Configured)  = Non-redundant
Redundancy State              = Non Redundant
```

In this example, HA for the dual RPs is not configured, as indicated by the nonredundant operational mode.

Step 3 Enter **configure terminal** to enter global configuration mode:

```
Router# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#
```

Step 4 Enter the **redundancy** command to enter redundancy configuration mode:

```
Router(config)# redundancy
Router(config-red)#
```

Step 5 Enter **mode sso** to enable SSO, or **mode none** to disable dual software redundancy:

```
Router(config-red)# mode sso

Router(config-red)# mode none
```

Step 6 Return to privileged EXEC mode using any method, such as entering **Ctrl-Z** or the **exit** command multiple times until you get to the # router prompt:

```
Router(config-red)#^Z
Router#

Router(config-red)#exit
Router(config)#exit
Router#
```

Step 7 Enter the **show redundancy states** command to confirm the configured redundancy mode changed to your new configuration.

In the following example, the configured redundancy mode has been changed to SSO. Note that the operation redundancy mode remains unchanged:

```
Router# show redundancy states
       my state = 13 -ACTIVE
     peer state = 1  -DISABLED
           Mode = Simplex
        Unit ID = 6

Redundancy Mode (Operational) = Non-redundant
Redundancy Mode (Configured)  = sso
Redundancy State              = Non Redundant
```

Step 8 Enter the **copy running-config startup-config** command to save the new configuration, and press **enter** to confirm the filename (or change the name, if desired):

```
Router# copy running-config startup-config
Destination filename [startup-config]?
Building configuration...
[OK]
PE25_ASR-1004#
```

Step 9 After the reload is complete, enter the **show redundancy states** command to confirm the operational redundancy mode has changed to the mode that you configured in Step 5.

In this example, the operational redundancy mode has changed to SSO:

```
Router# show redundancy states
       my state = 13 -ACTIVE
     peer state = 8  -STANDBY HOT
           Mode = Duplex
        Unit ID = 6

Redundancy Mode (Operational) = sso
Redundancy Mode (Configured)  = sso
Redundancy State              = sso
```

Step 10 The router must be reloaded for the new HA configuration. Enter the **reload** command to reload the router:

```
Router# reload
Proceed with reload? [confirm]
<bootup messages removed for brevity>
```

If you are not using the console port to complete this procedure, your session may be terminated at this point. If you session is terminated, give the router a few minutes to complete the reload and then log into the router.

Step 11 (Optional) To confirm an active and a standby IOS process exist, and how the processes are numbered, enter the **show platform** command.

In this example, the **show platform** command output reveals that R0/0 is the active IOS process and R0/1 is the standby IOS process:

```
Router# show platform
Chassis type: ASR1004

Slot      Type                State                 Insert time (ago)
--------- ------------------- --------------------- -----------------
0         ASR1000-SIP10       ok                    00:52:34
 0/0      SPA-2X1GE-V2        ok                    00:51:30
 0/1      SPA-2XT3/E3         ok                    00:51:24
R0        ASR1000-RP1         ok                    00:52:34
 R0/0                         ok, active            00:52:34
 R0/1                         ok, standby           00:51:00
F0        ASR1000-ESP10       ok, active            00:52:34
P0        ASR1004-PWR-AC      ok                    00:51:57
P1        ASR1004-PWR-AC      ok                    00:51:57
```

Using Subpackages for Software Upgrade on a Cisco ASR 1001 Router, Cisco ASR 1001-X Router or a Cisco ASR 1002-X Router

This section provides instructions on using software upgrade for Cisco ASR 1001 Router, Cisco ASR 1001-X Router or Cisco ASR 1002-X Router running subpackages.

These instructions assume two IOS processes are active on the RP and that the router is already running using subpackages.

SUMMARY STEPS

1. **show version**

   show version installed

   dir *filesystem*:<*directory*>

   show platform

   show redundancy-states
2. **redundancy**

   mode sso
3. **mkdir** *URL-to-directory-name*
4. **ip tftp** *source-interface gigabitethernet port*
5. **copy tftp:** *URL-to-target-location*
6. **request platform software package expand file** *URL-to-consolidated-package*
7. **dir** *URL-to-consolidated-package*
8. **copy** *file-system*:**asr1001-espbase.**_version_.**pkg** *URL-to-directory-of-sub-packages-active-RP*

   copy file-system:**asr1001-rpaccess**.version.**pkg** URL-to-directory-of-sub-packages-active-RP

   copy file-system:**asr1001-rpbase**.version.**pkg** URL-to-directory-of-sub-packages-active-RP

   copy file-system:**asr1001-rpcontrol**.version.**pkg** URL-to-directory-of-sub-packages-active-RP

   copy file-system:**asr1001-rpios**.version.**pkg** URL-to-directory-of-sub-packages-active-RP

   copy file-system:**asr1001-sipbase**.version.**pkg** URL-to-directory-of-sub-packages-active-RP

   copy file-system:**asr1001-sipspa**.version.**pkg** URL-to-directory-of-sub-packages-active-RP
9. **issu loadversion rp 0 file** *file-system*:**asr1001-{rpaccess,rpios,rpcontrol}***_version-string_***.pkg bay** *standby-bay* **force**
10. **issu commitversion**
11. **redundancy force-switchover**
12. **show platform**
13. **issu loadversion rp 0 file** *file-system*:**asr1001-{rpaccess,rpios,rpcontrol}***_version-string_***.pkg bay** *standby-bay* **force**
14. **issu commitversion**
15. **issu loadversion rp 0 file** *file-system*:**asr1001-{rpbase,sipbase,sipspa,espbase}***_version_***.pkg force**
16. **show version installed**
17. **reload**

Note Step 17 does not have to be performed immediately, and should be done at a convenient time.

DETAILED STEPS

	Command or Action	Purpose

Step 1	**show version**
	show version installed
	dir *filesystem:<directory>*
	show platform
	show redundancy-states
	
	Example:
	`Router# show version`
	`Router# show version r0 installed`
	`Router# dir bootflash:`
	`Router# show platform`
	`Router# show redundancy-states`
		(Optional) Use the following commands to confirm the current router configuration, as follows:
		• **show version** and **show version** *active-rp* **installed**—Verify the running version of the Cisco IOS XE software on the router, and which file was used to boot the router, and where that file is stored.
		• **dir**—Confirm that the files that were used to boot the router are located in the directory.
		• **show platform**—Confirm the current status of the active and standby RPs.
		• **show redundancy states**—Confirm the operational and configured redundancy states.

Step 2	**redundancy**
	mode sso
	
	Example:
	`Router(config)# redundancy`
	`Router(config-red)# mode sso`
		Configure SSO if it is not already configured.
		Note Save the configuration after making this configuration step.

Step 3	**mkdir** *URL-to-directory-name* **Example:** `Router# mkdir 221subs`	Create a directory to store the consolidated package and subpackages. This directory must be created in most cases because the consolidated packages and subpackages have to be separated from the subpackages that booted the router at this step of the procedure.
Step 4	**ip tftp** *source-interface gigabitethernet port* **Example:** `Router(config)# ip tftp source-interface gigabitethernet 0`	Specifies the Gigabit Ethernet TFTP source-interface to be configured: *slot/port*—Specifies the location of the TFTP source-interface. **Note** To copy a file using TFTP through the Management Ethernet interface, the **ip tftp source-interface GigabitEthernet 0** command must be entered before entering the **copy tftp** command.
Step 5	**copy tftp:** *URL-to-target-location* **Example:** `Router# copy tftp: 221subs`	Copy the consolidated package file into the directory created in Step 3. The consolidated package in this step should not be copied into the same directory where the subpackages that are currently running your router are stored (the directory containing the packages.conf provisioning file from which the router was booted). **Tip** It is recommended that you copy the package onto a usb: or harddisk: file system for space considerations when performing this step of the procedure.

	Command or Action	Purpose

Step 6	**request platform software package expand file** *URL-to-consolidated-package* *(Optional) dir target-URL*
	
	Example:
	`Router# request platform software package expand file usb0:221subs/asr1001-universalk9.03.02.01.S.151-1.S1_2.bin`
	`Router# dir 221subs`
		Extract the subpackages out of the consolidated package file into the temporary directory.
		
		Note Take extra care to extract the subpackages to a temporary subdirectory and do not delete any of the files currently running the router at this point of the procedure.
		
		To erase the files that were running on the router before the ISSU upgrade, enter the **request platform software package clean** command after the ISSU upgrade has been completed.

Step 7	**dir** *target-URL* **Example:** `Router# dir 221subs`	(Optional) Display the directory to confirm that the files were extracted.

Step 8	**copy** file-system:**asr1001-espbase**.version.**pkg** URL-to-directory-of-sub-packages-active-RP
	copy file-system:**asr1001-rpaccess**.version.**pkg** URL-to-directory-of-sub-packages-active-RP
	copy file-system:**asr1001-rpbase**.version.**pkg** URL-to-directory-of-sub-packages-active-RP
	copy file-system:**asr1001-rpcontrol**.version.**pkg** URL-to-directory-of-sub-packages-active-RP
	copy file-system:**asr1001-rpios**.version.**pkg** URL-to-directory-of-sub-packages-active-RP
	copy file-system:**asr1001-sipbase**.version.**pkg** URL-to-directory-of-sub-packages-active-RP
	copy file-system:**asr1001-sipspa**.version.**pkg** URL-to-directory-of-sub-packages-active-RP
	
	Example:
	`Router# copy 221subs/asr1001-espbase.03.02.01.S.151-1.S1_2.pkg bootflash:`
	`Router# copy 221subs/asr1001-rpaccess.03.02.01.S.151-1.S1_2.pkg bootflash:`
	`Router# copy 221subs/asr1001-rpbase.03.02.01.S.151-1.S1_2.pkg bootflash:`
	`Router# copy 221subs/asr1001-rpcontrol.03.02.01.S.151-1.S1_2.pkg bootflash:`
	`Router# copy 221subs/asr1001-rpios-universalk9.03.02.01.S.151-1.S1_2.pkg bootflash:`
	`Router# copy 221subs/asr1001-sipbase.03.02.01.S.151-1.S1_2.pkg bootflash:`
	`Router# copy 221subs/asr1001-sipspa.03.02.01.S.151-1.S1_2.pkg bootflash:`
		Copy the subpackages out of the temporary directory into the directory on the router where the subpackages running the active RP are currently stored.

	Command or Action	Purpose
Step 9	**issu loadversion rp 0 file** *file-system*:**asr1001-{rpaccess,rpios,rpcontrol}*** *version-string*.**pkg bay** *standby-bay* **force** **Example:** `Router# issu loadversion rp 0 file bootflash:asr1001-{rpaccess,rpios,rpcontrol}*03.02.01.S.151-1.S1_2*.pkg bay 1 force`	Upgrade the RPAccess, RPIOS, and RPControl subpackages in the standby bay.
Step 10	**issu commitversion** **Example:** `Router# issu commitversion`	Once the SSO state is reached, commit the software version.
Step 11	**redundancy force-switchover** **Example:** `Router# redundancy force-switchover`	Force a switchover from the active IOS process to the standby IOS process. **Note** Your connection to the router often drops and is expected behavior at this point of the procedure in many scenarios. If this step drops your connection to the router, wait a few minutes before reconnecting to the router and then continue to Step 12.
Step 12	**show platform** **Example:** `Router# show platform`	(Optional) Monitor system state to ensure both IOS processes are active.
Step 13	**issu loadversion rp 0 file** *file-system*:**asr1001-{rpaccess,rpios,rpcontrol}*** *version-string*.**pkg bay** *standby-bay* **force** **Example:** `Router# issu loadversion rp 0 file bootflash:asr1001-{rpaccess,rpios,rpcontrol}*03.02.01.S.151-1.S1_2*.pkg bay 0 force`	Upgrade the RPAccess, RPIOS, and RPControl subpackages in the standby bay (a different bay than in Step 9).
Step 14	**issu commitversion** **Example:** `Router# issu commitversion`	Commit the software version.

	Command or Action	Purpose

Step 15	**For ASR 1001 and ASR 1002-X**
	issu loadversion rp 0 file *file-system*:**asr1001-{rpbase,sipbase,sipspa,espbase}*** *version*.**pkg force**
	
	Example:
	`Router#issu loadversion rp 0 file bootflash:221subs/asr1001-{rpbase,sipbase,sipspa,espbase}*03.02.01.S.151-1.S1_2.pkg force`
	
	For ASR 1001-X
	issu loadversion rp 0 file *file-system*:**asr1001-{rpbase,sipbase,sipspa,esp,firmware}*** *version*.**pkg force**
	
	Example:
	`Router#issu loadversion rp 0 file bootflash:221subs/asr1001-{rpbase,sipbase,sipspa,esp,firmware}*03.02.01.S.151-1.S1_2.pkg force`
		Upgrade the RP, SIP, SPA, and ESP subpackages for each SIP on the router.
		
		Note This step must be completed one SIP at a time, and repeated for each SIP installed on the router before performing the next step.
		
		Tip You can use the **show ip interface brief** command to identify which slots contain SIPs and SPAs. The interfaces with three numbers (in the form *SIP-number*/*SPA-number*/*interface-number*) identify the SIP and SPA locations in the router.
		
		Note The *pattern* options used in this CLI (**rpbase**, **sipbase**, **sipspa, espbase** and **firmware**) were introduced in Cisco IOS XE Release 2.1.2 and are not available in previous Cisco IOS XE Releases.
		See the "ISSU Procedures (Prior to Cisco IOS XE Release 2.1.2)" section on page 5-72 for pre-Cisco IOS XE Release 2.1.2 ISSU upgrade procedures.

Step 16	**show version installed** **Example:** `Router# show version installed`	(Optional) Verify that the subpackages are properly installed.
Step 17	**reload** **Example:** `Router# reload`	(Optional) Reload the RP. **Tip** The router will continue normal operation even without a reload, so you can reload the router during scheduled maintenance or a slower traffic period. It is highly recommended reload to avoid conflicts in the software base.

Examples

The following example shows the software upgrade for Cisco ASR 1001 Router, Cisco ASR 1001-X Router or Cisco ASR 1002-X Router running subpackages.

```
Router#show version
Cisco IOS Software, IOS-XE Software (X86_64_LINUX_IOSD-UNIVERSALK9-M), Experimental
Version 15.1(20110301:124851) [asr1001-universalk9.03.02.01.S.151-1.S1]
Copyright (c) 1986-2011 by Cisco Systems, Inc.
Compiled Tue 01-Mar-11 06:25 by mcpre


Cisco IOS-XE software, Copyright (c) 2005-2011 by cisco Systems, Inc.
All rights reserved.  Certain components of Cisco IOS-XE software are
licensed under the GNU General Public License ("GPL") Version 2.0.  The
software code licensed under GPL Version 2.0 is free software that comes
with ABSOLUTELY NO WARRANTY.  You can redistribute and/or modify such
GPL code under the terms of GPL Version 2.0.  For more details, see the
documentation or "License Notice" file accompanying the IOS-XE software,
or the applicable URL provided on the flyer accompanying the IOS-XE
```

```
software.


ROM: IOS-XE ROMMON

Router uptime is 5 days, 26 minutes
Uptime for this control processor is 5 days, 27 minutes
System returned to ROM by reload
System image file is "bootflash:asr1001-universalk9.03.02.01.S.151-1.S1"
Last reload reason: Reload Command



This product contains cryptographic features and is subject to United
States and local country laws governing import, export, transfer and
use. Delivery of Cisco cryptographic products does not imply
third-party authority to import, export, distribute or use encryption.
Importers, exporters, distributors and users are responsible for
compliance with U.S. and local country laws. By using this product you
agree to comply with applicable laws and regulations. If you are unable
to comply with U.S. and local laws, return this product immediately.

A summary of U.S. laws governing Cisco cryptographic products may be found at:
http://www.cisco.com/wwl/export/crypto/tool/stqrg.html

If you require further assistance please contact us by sending email to
export@cisco.com.


License Info:

License UDI:
Device# PID                     SN                      UDI
-------------------------------------------------------------------------------
*0      ASR1001                 JAE14020AT1             ASR1001:JAE14020AT1


License Package Information for Module:'asr1001'

Module name   Image level          Pri   Config Valid license
-----------------------------------------------------------------------
asr1001       adventerprise        1     NO     adventerprise
              advipservices        2     NO     advipservices
              ipbase               3     NO     ipbase

Module name   Current Level        Reboot Level
-----------------------------------------------------------------------
asr1001       ipbase               ipbase

cisco ASR1001 (1RU) processor with 3851680K/6147K bytes of memory.
6 Gigabit Ethernet interfaces
32768K bytes of non-volatile configuration memory.
8388608K bytes of physical memory.
7782399K bytes of eUSB flash at bootflash:.

Configuration register is 0x8102

Router# show version installed
Package: Provisioning File, version: n/a, status: active
  File: consolidated:packages.conf, on: RP0
  Built: n/a, by: n/a
  File SHA1 checksum: e11c9aacb8c233ed10eeac61ca20a9469fb0b81e

Package: rpbase, version: 03.02.01.S.151-1.S1, status: active
```

```
  File: consolidated:asr1001-rpbase.03.02.01.S.151-1.S1.pkg, on: RP0
  Built: 2011-03-01_07.10, by: mcpre
  File SHA1 checksum: e3c93ac457f2632ae90ba346692d58001cf0d02c

Package: rpcontrol, version: 03.02.01.S.151-1.S1, status: active
  File: consolidated:asr1001-rpcontrol.03.02.01.S.151-1.S1.pkg, on: RP0/0
  Built: 2011-03-01_07.10, by: mcpre
  File SHA1 checksum: 59ba43488cf2a0e7bc06b66fad0e574bbdf42a62
<output removed for brevity>


Router#show redundancy states
       my state = 13 -ACTIVE
     peer state = 1  -DISABLED
           Mode = Simplex
           Unit = Primary
        Unit ID = 48

Redundancy Mode (Operational) = Non-redundant
Redundancy Mode (Configured)  = Non-redundant
Redundancy State              = Non Redundant
    Manual Swact = disabled (system is simplex (no peer unit))
 Communications = Down      Reason: Simplex mode

   client count = 63
 client_notification_TMR = 30000 milliseconds
          keep_alive TMR = 4000 milliseconds
        keep_alive count = 0
    keep_alive threshold = 7
           RF debug mask = 0x0

Router# show platform
Chassis type: ASR1001

Slot      Type                State                 Insert time (ago)
--------- ------------------- --------------------- -----------------
0         ASR1001             ok                    5d07h
 0/0      ASR1001             ok                    5d07h
 0/1      SPA-2X1GE-V2        ok                    5d07h
R0        ASR1001             ok, active            5d07h
F0        ASR1001             ok, active            5d07h
P0        ASR1001-PWR-AC      ps, fail              5d07h
P1        ASR1001-PWR-AC      ok                    5d07h
P2        ASR1001-FANTRAY     ok                    5d07h

Slot      CPLD Version        Firmware Version
--------- ------------------- ----------------------------------------
0         1001050E            12.2(20090526:143323) [gschnorr-mcp_...
R0        10010514            12.2(20090526:143323) [gschnorr-mcp_...
F0        1001050E            12.2(20090526:143323) [gschnorr-mcp_...

Router# mkdir 221subs
Create directory filename [221subs]?
Created dir bootflash:221subs

Router(config)# ip tftp source-interface gigabitethernet 0
Router# copy tftp: 221subs
Address or name of remote host []? 172.27.55.254
Source filename []? /auto/users/asr1001-universalk9.03.02.01.S.151-1.S1_2.bin,12
Destination filename [asr1001-universalk9.03.02.01.S.151-1.S1.bin,12]?
Accessing
tftp://172.27.55.254//auto/users/asr1001-universalk9.03.02.01.S.151-1.S1_2.bin,12...
```

```
Loading /auto/users/asr1001-universalk9.03.02.01.S.151-1.S1.bin,12 (via GigabitEthernet0):
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
```

```
Copy in
progress...CCCC
```

```
--- Starting local lock acquisition on R0 ---
Finished local lock acquisition on R0

--- Starting file path checking ---
Finished file path checking

--- Starting image file verification ---
Checking image file names
Locating image files and validating name syntax
  Found asr1001-rpaccess.03.02.01.S.151-1.S1_2.pkg
  Found asr1001-rpios-universalk9.03.02.01.S.151-1.S1_2.pkg
  Found asr1001-rpcontrol.03.02.01.S.151-1.S1_2.pkg
Verifying image file locations
Inspecting image file types
Processing image file constraints
Creating candidate provisioning file
Finished image file verification

--- Starting candidate package set construction ---
Verifying existing software set
Processing candidate provisioning file
Constructing working set for candidate package set
Constructing working set for running package set
Checking command output
Constructing merge of running and candidate packages
Checking if resulting candidate package set would be complete
Finished candidate package set construction

--- Starting compatibility testing ---
Determining whether candidate package set is compatible

WARNING:
WARNING: Candidate software combination not found in compatibility database
WARNING:

Determining whether installation is valid

WARNING:
WARNING: Candidate software combination not found in compatibility database
WARNING:


WARNING:
WARNING: Candidate software combination not found in compatibility database
WARNING:

Software sets are identified as compatible
Verifying image type compatibility
Checking IPC compatibility with running software
Checking candidate package set infrastructure compatibility
Checking infrastructure compatibility with running software
Checking package specific compatibility
Finished compatibility testing

--- Starting impact testing ---
Checking operational impact of change
    WARNING: Connection may be lost during installation of IOS package
Finished impact testing

--- Starting list of software package changes ---
No old package files removed
New files list:
  Added asr1001-rpaccess.03.02.01.S.151-1.S1_2.pkg
  Added asr1001-rpcontrol.03.02.01.S.151-1.S1_2.pkg
```

```
  Added asr1001-rpios-universalk9.03.02.01.S.151-1.S1_2.pkg
Finished list of software package changes

--- Starting commit of software changes ---
Updating provisioning rollback files
Creating pending provisioning file
Committing provisioning file
Finished commit of software changes

--- Starting analysis of software changes ---
Finished analysis of software changes

--- Starting update running software ---
Blocking peer synchronization of operating information
Creating the command set placeholder directory
  Finding latest command set
  Finding latest command shortlist lookup file
  Finding latest command shortlist file
  Assembling CLI output libraries
  Assembling CLI input libraries
  Assembling Dynamic configuration files
  Applying interim IPC and database definitions
  Replacing running software
  Replacing CLI software
  Restarting software
  Restarting IOS PID: 9275, in slot/bay 0/1

*Mar 15 16:28:50.014: %REDUNDANCY-3-STANDBY_LOST: Standby processor fault
(PEER_NOT_PRESENT)
*Mar 15 16:28:50.014: %REDUNDANCY-3-STANDBY_LOST: Standby processor fault (PEER_DOWN)
*Mar 15 16:28:50.014: %REDUNDANCY-3-STANDBY_LOST: Standby processor fault
(PEER_REDUNDANCY_STATE_CHANGE)
*Mar 15 16:28:52.224: %RF-5-RF_RELOAD: Peer reload. Reason: EHSA standby down
*Mar 15 16:29:29.214: %REDUNDANCY-5-PEER_MONITOR_EVENT: Active detected a standby
insertion (raw-event=PEER_FOUND(4))
*Mar 15 16:29:29.214: %REDUNDANCY-5-PEER_MONITOR_EVENT: Active detected a standby
insertion (raw-event=PEER_REDUNDANCY_STATE_CHANGE(5))
  Generating software version information

  Unblocking peer synchronization of operating information
Unmounting old packages
Cleaning temporary installation files
  Finished update running software


Router#issu commitversion
--- Starting local lock acquisition on R0 ---
Finished local lock acquisition on R0

--- Starting installation changes ---
Cancelling rollback timer
Finished installation changes

SUCCESS: Installation changes committed
Router#
*Mar 15 16:30:24.885: %NBAR_HA-5-NBAR_INFO: NBAR sync DONE!
*Mar 15 16:30:25.421: %HA_CONFIG_SYNC-6-BULK_CFGSYNC_SUCCEED: Bulk Sync succeedd
*Mar 15 16:30:25.423: %RF-5-RF_TERMINAL_STATE: Terminal state reached for (SSO)

Router#redundancy force-switchover
Proceed with switchover to standby RP? [confirm]y    Manual Swact = enabled


%ASR1000_INFRA-6-CONSOLE_ACTIVE: R0/1 console active.  Press RETURN to get star!
```

```
*Mar 15 16:31:00.014: %REDUNDANCY-3-SWITCHOVER: RP switchover (PEER_NOT_PRESENT)
*Mar 15 16:31:00.015: %REDUNDANCY-3-SWITCHOVER: RP switchover (PEER_DOWN)
*Mar 15 16:31:00.015: %REDUNDANCY-3-SWITCHOVER: RP switchover (PEER_REDUNDANCY_)
*Mar 15 16:31:00.328: %LINK-3-UPDOWN: Interface Lsmpi0, changed state to up
*Mar 15 16:31:00.328: %LINK-3-UPDOWN: Interface EOBC0, changed state to up
*Mar 15 16:31:00.328: %LINK-3-UPDOWN: Interface LIIN0, changed state to up
*Mar 15 16:31:01.328: %LINEPROTO-5-UPDOWN: Line protocol on Interface Lsmpi0, cp
*Mar 15 16:31:01.328: %LINEPROTO-5-UPDOWN: Line protocol on Interface EOBC0, chp
*Mar 15 16:31:01.328: %LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEtn
*Mar 15 16:31:01.328: %LINEPROTO-5-UPDOWN: Line protocol on Interface LIIN0, chp
*Mar 15 16:31:02.064: %RF-5-RF_RELOAD: Self reload. Reason: redundancy force-swr
*Mar 15 16:31:02.293: %SYS-5-SWITCHOVER: Switchover requested by red_switchover.
*Mar 15 16:31:02.321: %LINK-3-UPDOWN: Interface Null0, changed state to up
*Mar 15 16:31:03.321: %LINEPROTO-5-UPDOWN: Line protocol on Interface Null0, chp
*Mar 15 16:31:03.322: %LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEtn
*Mar 15 16:31:03.322: %LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEtn
*Mar 15 16:31:03.322: %LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEtn
*Mar 15 16:31:03.322: %LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEtn
*Mar 15 16:31:03.322: %LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEtn
*Mar 15 16:31:03.322: %LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEtn
*Mar 15 16:31:43.189: %REDUNDANCY-5-PEER_MONITOR_EVENT: Active detected a stand)
*Mar 15 16:31:43.189: %REDUNDANCY-5-PEER_MONITOR_EVENT: Active detected a stand)
*Mar 15 16:32:28.411: %NBAR_HA-5-NBAR_INFO: NBAR sync DONE!
*Mar 15 16:32:28.971: %HA_CONFIG_SYNC-6-BULK_CFGSYNC_SUCCEED: Bulk Sync succeedd
*Mar 15 16:32:28.973: %RF-5-RF_TERMINAL_STATE: Terminal state reached for (SSO)
```

```
Router# issu loadversion rp 0 file
bootflash:221subs/asr1001-{rpaccess,rpios,rpcontrol}*03.02.01.S.151-1.S1_2.pkg bay 0 force
--- Starting local lock acquisition on R0 ---
Finished local lock acquisition on R0

--- Starting file path checking ---
Finished file path checking

--- Starting image file verification ---
Checking image file names
Locating image files and validating name syntax
Verifying image file locations
Inspecting image file types
Processing image file constraints
Creating candidate provisioning file
Finished image file verification

--- Starting candidate package set construction ---
Verifying existing software set
Processing candidate provisioning file
Constructing working set for candidate package set
Constructing working set for running package set
Checking command output
Constructing merge of running and candidate packages
Checking if resulting candidate package set would be complete
Finished candidate package set construction

--- Starting compatibility testing ---
Determining whether candidate package set is compatible

WARNING:
WARNING: Candidate software combination not found in compatibility database
WARNING:

Determining whether installation is valid
Software sets are identified as compatible
```

```
Verifying image type compatibility
Checking IPC compatibility with running software
Checking candidate package set infrastructure compatibility
Checking infrastructure compatibility with running software
Checking package specific compatibility
Finished compatibility testing

--- Starting impact testing ---
Checking operational impact of change
    WARNING: Connection may be lost during installation of IOS package
Finished impact testing

--- Starting list of software package changes ---
Old files list:
  Removed asr1001-rpaccess.03.02.01.S.151-1.S1.pkg
  Removed asr1001-rpcontrol.03.02.01.S.151-1.S1.pkg
  Removed asr1001-rpios-universalk9.03.02.01.S.151-1.S1.pkg
No new package files added
Finished list of software package changes

--- Starting commit of software changes ---
Updating provisioning rollback files
Creating pending provisioning file
Committing provisioning file
Finished commit of software changes

--- Starting analysis of software changes ---
Finished analysis of software changes

--- Starting update running software ---
Blocking peer synchronization of operating information
Creating the command set placeholder directory
  Finding latest command set
  Finding latest command shortlist lookup file
  Finding latest command shortlist file
  Assembling CLI output libraries
  Assembling CLI input libraries
  Assembling Dynamic configuration files
  Applying interim IPC and database definitions
  Replacing running software
  Replacing CLI software
  Restarting software
  Restarting IOS PID: 20548, in slot/bay 0/0

*Mar 15 16:36:14.365: %REDUNDANCY-3-STANDBY_LOST: Standby processor fault
(PEER_NOT_PRESENT)
*Mar 15 16:36:14.365: %REDUNDANCY-3-STANDBY_LOST: Standby processor fault (PEER_DOWN)
*Mar 15 16:36:14.365: %REDUNDANCY-3-STANDBY_LOST: Standby processor fault
(PEER_REDUNDANCY_STATE_CHANGE)
*Mar 15 16:36:16.558: %RF-5-RF_RELOAD: Peer reload. Reason: EHSA standby down
*Mar 15 16:36:49.365: %REDUNDANCY-5-PEER_MONITOR_EVENT: Active detected a standby
insertion (raw-event=PEER_FOUND(4))
*Mar 15 16:36:49.365: %REDUNDANCY-5-PEER_MONITOR_EVENT: Active detected a standby
insertion (raw-event=PEER_REDUNDANCY_STATE_CHANGE(5))
Notifying running software of updates
  Unblocking peer synchronization of operating information
Unmounting old packages
Cleaning temporary installation files
  Finished update running software

SUCCESS: Finished installing software.


Router#issu commitversion
```

```
--- Starting local lock acquisition on R0 ---
Finished local lock acquisition on R0

--- Starting installation changes ---
Cancelling rollback timer
Finished installation changes
SUCCESS: Installation changes committed


Router#issu loadversion rp 0 file
bootflash:221subs/asr1001-{rpbase,sipbase,sipspa,espbase}*03.02.01.S.151-1.S1_2.pkg force
--- Starting local lock acquisition on R0 ---
Finished local lock acquisition on R0

--- Starting file path checking ---
Finished file path checking

--- Starting image file verification ---
Checking image file names
Locating image files and validating name syntax
Verifying image file locations
Inspecting image file types
    WARNING: In-service installation of RP Base package
    WARNING: requires software reboot of target RP
    WARNING: Automatically setting the on-reboot flag
Processing image file constraints
Creating candidate provisioning file
Finished image file verification

--- Starting candidate package set construction ---
Verifying existing software set
Processing candidate provisioning file
Constructing working set for candidate package set
Constructing working set for running package set
Checking command output
Constructing merge of running and candidate packages
Checking if resulting candidate package set would be complete
Finished candidate package set construction

--- Starting compatibility testing ---
Determining whether candidate package set is compatible
Determining whether installation is valid
Determining whether installation is valid ... skipped
Verifying image type compatibility
Checking IPC compatibility for candidate software
Checking candidate package set infrastructure compatibility
Checking infrastructure compatibility with running software
Checking infrastructure compatibility with running software ... skipped
Checking package specific compatibility
Finished compatibility testing

--- Starting list of software package changes ---
Old files list:
  Removed asr1001-espbase.03.02.01.S.151-1.S1.pkg
  Removed asr1001-rpbase.03.02.01.S.151-1.S1.pkg
  Removed asr1001-sipbase.03.02.01.S.151-1.S1.pkg
  Removed asr1001-sipspa.03.02.01.S.151-1.S1.pkg
New files list:
  Added asr1001-espbase.03.02.01.S.151-1.S1_2.pkg
  Added asr1001-rpbase.03.02.01.S.151-1.S1_2.pkg
  Added asr1001-sipbase.03.02.01.S.151-1.S1_2.pkg
  Added asr1001-sipspa.03.02.01.S.151-1.S1_2.pkg
Finished list of software package changes
```

```
--- Starting commit of software changes ---
Updating provisioning rollback files
Creating pending provisioning file
Committing provisioning file
Finished commit of software changes

SUCCESS: Software provisioned.  New software will load on reboot.

Router# reload
```

Using Subpackages for Software Upgrade on a Cisco ASR 1002 Router or Cisco ASR 1004 Router (software upgrade Command Set)

This section provides instructions on using software upgrade for Cisco ASR 1002 or 1004 Router running subpackages.

These instructions assume two IOS processes are active on the RP and that the router is already running using subpackages. For information on checking and configuring two IOS processes on the same RP, see the "Using Subpackages for Software Upgrade on a Cisco ASR 1002 Router or Cisco ASR 1004 Router (software upgrade Command Set)" section on page 5-91.

SUMMARY STEPS

1. **show version**

   show version installed

   dir *filesystem*:<*directory*>

   show platform

   show redundancy-states
2. **redundancy**

   mode sso
3. **mkdir** *URL-to-directory-name*
4. **ip tftp** *source-interface gigabitethernet port*
5. **copy tftp:** *URL-to-target-location*
6. **request platform software package expand file** *URL-to-consolidated-package*
7. **dir** *URL-to-consolidated-package*
8. **copy** *file-system*:**asr1000rp2-espbase.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*

   copy *file-system*:**asr1000rp2-espx86base.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*

   copy *file-system*:**asr1000rp2-rpaccess.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*

   copy *file-system*:**asr1000rp2-rpbase.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*

   copy *file-system*:**asr1000rp2-rpcontrol.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*

   copy *file-system*:**asr1000rp2-rpios.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*

   copy *file-system*:**asr1000rp2-sipbase.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*

   copy *file-system*:**asr1000rp2-sipspa.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*

   copy *file-system*:**asr1000rp2-elcbase.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*

copy *file-system*:**asr1000rp2-elcspa.**_version_.**pkg** *URL-to-directory-of-sub-packages-active-RP*

Note In step 8, each individual subpackage that was extracted in step 6 is copied to the directory where the subpackages that are currently running the active RP are stored.

9. **issu loadversion rp 0 file** *file-system*:**asr1000rp2-{rpaccess,rpios,rpcontrol}***_version-string_***.pkg bay** *standby-bay* **force**

   issu commitversion
10. **redundancy force-switchover**
11. **show platform**
12. **issu loadversion rp 0 file** *file-system*:**asr1000rp2-{rpaccess,rpios,rpcontrol}***_version-string_***.pkg bay** *standby-bay* **force**

    issu commitversion
13. **issu loadversion rp 0 file** *file-system*:**asr1000rp2-{sipbase,sipspa}***_version_***.pkg slot** *SIP-slot-number* **force**

    issu commitversion

    Repeat the step 13, for each available SIP installed in the router before moving onto the next step.
14. **issu loadversion rp 0 file** *file-system*:**asr1000rp2-{elcbase,elcspa}***_version_***.pkg slot** *ELCslot-number* **force**

    issu commitversion

    Repeat the step 14, for each available ELC installed in the router before moving onto the next step.
15. **issu loadversion rp 0 file** *file-system*:**asr1000rp2-esp***_version_***.pkg force**

    issu commitversion
16. **issu loadversion rp 0 file** *file-system*:**asr1000rp***_version_***.pkg**

    issu commitversion
17. **show version installed**
18. **reload**
19. **request platform software package clean**

Note Step 17, 18, and 19 does not have to be performed immediately, and can be done at a convenient time.

DETAILED STEPS

	Command or Action	Purpose

Step 1	**show version**
	show version installed
	dir *filesystem:<directory>*
	show platform
	show redundancy-states
	
	Example:
	`Router# show version`
	`Router# show version r0 installed`
	`Router# dir bootflash:`
	`Router# show platform`
	`Router# show redundancy-states`
		(Optional) Use the following commands to confirm the current router configuration, as follows:
		• **show version** and **show version** *active-rp* **installed**—Verify the running version of the Cisco IOS XE software on the router, and which file was used to boot the router, and where that file is stored.
		• **dir**—Confirm that the files that were used to boot the router are located in the directory.
		• **show platform**—Confirm the current status of the active and standby RPs.
		• **show redundancy states**—Confirm the operational and configured redundancy states.

Step 2	**redundancy**
	mode sso
	
	Example:
	`Router(config)# redundancy`
	`Router(config-red)# mode sso`
		Configure SSO if it is not already configured.
		Note Save the configuration after making this configuration step.

Step 3	**mkdir** *URL-to-directory-name* **Example:** `Router# mkdir usb0:221subs`	Create a directory to store the consolidated package and subpackages. This directory must be created in most cases because the consolidated packages and subpackages have to be separated from the subpackages that booted the router at this step of the procedure.
Step 4	**ip tftp** *source-interface gigabitethernet port* **Example:** `Router(config)# ip tftp source-interface gigabitethernet 0`	Specifies the Gigabit Ethernet TFTP source-interface to be configured: *slot/port*—Specifies the location of the TFTP source-interface. **Note** To copy a file using TFTP through the Management Ethernet interface, the **ip tftp source-interface GigabitEthernet 0** command must be entered before entering the **copy tftp** command.
Step 5	**copy tftp:** *URL-to-target-location* **Example:** `Router# copy tftp: usb0:221subs`	Copy the consolidated package file into the directory created in Step 3. The consolidated package in this step should not be copied into the same directory where the subpackages that are currently running your router are stored (the directory containing the packages.conf provisioning file from which the router was booted). **Tip** It is recommended that you copy the package onto a usb: or harddisk: file system for space considerations when performing this step of the procedure.

	Command or Action	Purpose

Step 6	**request platform software package expand file** *URL-to-consolidated-package*
	(Optional) **dir** *target-URL*
	
	Example:
	`Router# request platform software package expand file usb0:221subs/asr1000rp2-adventerprisek9.03.13.00.S.154-3.S-ext.bin`
	`Router# dir usb0:221subs`
		Extract the subpackages out of the consolidated package file into the temporary directory.
		
		Note Take extra care to extract the subpackages to a temporary subdirectory and do not delete any of the files currently running the router at this point of the procedure.
		
		To erase the files that were running on the router before the ISSU upgrade, enter the **request platform software package clean** command after the ISSU upgrade has been completed.

Step 7	**dir** *target-URL* **Example:** `Router# dir usb0:221subs`	(Optional) Display the directory to confirm that the files were extracted.

Command or Action	Purpose

Step 8 **copy** *file-system*:**asr1000rp2-espbase.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*
copy *file-system*:**asr1000rp2-espx86base.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*
copy *file-system*:**asr1000rp2-rpaccess.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*
copy *file-system*:**asr1000rp2-rpbase.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*
copy *file-system*:**asr1000rp2-rpcontrol.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*
copy *file-system*:**asr1000rp2-rpios.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*
copy *file-system*:**asr1000rp2-sipbase.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*
copy *file-system*:**asr1000rp2-sipspa.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*
copy *file-system*:**asr1000rp2-elcbase.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*
copy *file-system*:**asr1000rp2-elcspa.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*
 	
Example:
`Router# copy usb0:221subs/asr1000rp2-espbase.03.13.00.S.154-3.S-ext.pkg bootflash:`
 	
`Router# copy usb0:221subs/asr1000rp2-espx86base.03.13.00.S.154-3.S-ext.pkg bootflash:`
 	
`Router# copy usb0:221subs/asr1000rp2-rpaccess.03.13.00.S.154-3.S-ext.pkg bootflash:`
 	
`Router# copy usb0:221subs/asr1000rp2-rpbase.03.13.00.S.154-3.S-ext.pkg bootflash:`
 	
`Router# copy usb0:221subs/asr1000rp2-rpcontrol.03.13.00.S.154-3.S-ext.pkg bootflash:`
 	
`Router# copy usb0:221subs/asr1000rp2-rpios-adventerprisek9.03.13.00.S.154-3.S-ext.pkg bootflash:`
 	
`Router# copy usb0:221subs/asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg bootflash:`
 	
`Router# copy usb0:221subs/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg bootflash:`
 	
`Router# copy usb0:221subs/asr1000rp2-elcbase.03.13.00.S.154-3.S-ext.pkg bootflash:`
 	
`Router# copy usb0:221subs/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg bootflash:`
	Copy the subpackages out of the temporary directory into the directory on the router where the subpackages running the active RP are currently stored.
	
	Note Make sure to copy all the subpackage files extracted in step 6 to the current active packages directory.

	Command or Action	Purpose
Step 9	**issu loadversion rp 0 file** *file-system*:**asr1000rp2-{rpaccess,rpios,rpcontrol}*** *version-string*.**pkg bay** *standby-bay* **force** **Example:** `Router# issu loadversion rp 0 file bootflash:asr1000rp2-{rpaccess,rpios,rpcontrol}*03.13.00.S.154-3.S-ext*.pkg bay 1 force`	Upgrade the RPAccess, RPIOS, and RPControl subpackages in the standby bay.
Step 10	**issu commitversion** **Example:** `Router# issu commitversion`	Once the SSO state is reached, commit the software version.
Step 11	**redundancy force-switchover** **Example:** `Router# redundancy force-switchover`	Force a switchover from the active IOS process to the standby IOS process. **Note** Your connection to the router often drops and is expected behavior at this point of the procedure in many scenarios. If this step drops your connection to the router, wait a few minutes before reconnecting to the router and then continue to Step 12.
Step 12	**show platform** **Example:** `Router# show platform`	(Optional) Monitor system state to ensure both IOS processes are active.
Step 13	**issu loadversion rp 0 file** *file-system*:**asr1000rp2-{rpaccess,rpios,rpcontrol}*** *version-string*.**pkg bay** *standby-bay* **force** **Example:** `Router# issu loadversion rp 0 file bootflash:asr1000rp2-{rpaccess,rpios,rpcontrol}*03.13.00.S.154-3.S-ext*.pkg bay 0 force`	Upgrade the RPAccess, RPIOS, and RPControl subpackages in the standby bay (a different bay than in Step 9).
Step 14	**issu commitversion** **Example:** `Router# issu commitversion`	Commit the software version.

	Command or Action	Purpose
Step 15	**issu loadversion rp 0 file** *file-system*:**asr1000rp2-{sipbase,sipspa}***version***.pkg slot** *SIP-slot-number* **force** **issu commitversion** **Example:** Router# **issu loadversion rp 0 file bootflash:asr1000rp2-{sipspa,sipbase}*03.13.00.S.154-3.S-ext*.pkg slot 0 force** Router# **issu commitversion** Router# **issu loadversion rp 0 file bootflash:asr1000rp2-{sipspa,sipbase}*03.13.00.S.154-3.S-ext*.pkg slot 1 force** Router# **issu commitversion** Router# **issu loadversion rp 0 file bootflash:asr1000rp2-{sipspa,sipbase}*03.13.00.S.154-3.S-ext*.pkg slot 2 force** Router# **issu commitversion**	Upgrade the SIP and SPA subpackages for each SIP on the router. **Note** This step must be completed one SIP at a time, and repeated for each SIP installed on the router before performing the next step. **Tip** You can use the **show ip interface brief** command to identify which slots contain SIPs and SPAs. The interfaces with three numbers (in the form *SIP-number*/*SPA-number*/*interface-number*) identify the SIP and SPA locations in the router. **Note** The *pattern* options used in this CLI (**sipbase** and **sipspa**) were introduced in Cisco IOS XE Release 2.1.2 and are not available in previous Cisco IOS XE Releases. See the "ISSU Procedures (Prior to Cisco IOS XE Release 2.1.2)" section on page 5-72 for pre-Cisco IOS XE Release 2.1.2 ISSU upgrade procedures. **Note** See the "SUMMARY STEPS" section on page 5-91 for ASR1000 Fixed Ethernet Line Card (ELC) Upgrade.
Step 16	**issu loadversion rp 0 file** *file-system*:**asr1000rp2-{elcbase,elcspa}***version***.pkg slot** *SIP-slot-number* **force** **issu commitversion** Repeat this step for each ELC installed in your router before proceeding to the next step. **Example:** Router# **issu loadversion rp 0 file bootflash:asr1000rp2-{elcspa,elcbase}*03.13.00.S.154-3.S-ext*.pkg slot 0 force** Router# **issu commitversion** Router# **issu loadversion rp 0 file bootflash:asr1000rp2-{elcspa,elcbase}*03.13.00.S.154-3.S-ext*.pkg slot 1 force** Router# **issu commitversion** Router# **issu loadversion rp 0 file bootflash:asr1000rp2-{elcspa,elcbase}*03.13.00.S.154-3.S-ext*.pkg slot 2 force** Router# **issu commitversion**	Upgrade the ELC and SPA subpackages for each ELC on the router. **Note** This step must be completed for one ELC at a time, and repeated for each ELC installed on the router before performing the next step. **Tip** You can use the **show ip interface brief** command to identify which slots contain ELCs and SPAs. The interfaces with three numbers (in the form *ELC-number*/*SPA-number*/*interface-number*) identify the ELC and SPA locations in the router. **Note** The *pattern* options used in this CLI (**elcbase** and **elcspa**) were introduced in Cisco IOS XE Release 3.10Sand are not available in previous Cisco IOS XE Releases.

	Command or Action	Purpose

Step 17	**issu loadversion rp 0 file** *file-system*:**asr1000rp2-esp***version***.pkg force**
	issu commitversion
	
	Example:
	`Router# issu loadversion rp 0 file bootflash:asr1000rp2-esp*03.13.00.S.154-3.S-ext *.pkg force`
	`Router# issu commitversion`
		Upgrades the ESP Base subpackage and Commits the ESP Base software.

Step 18	**issu loadversion rp 0 file** *file-system*:**asr1000rp***version***.pkg**
	issu commitversion
	
	Example:
	`Router# issu loadversion rp 0 file bootflash:asr1000rp*03.13.00.S.154-3.S-ext*.pkg`
	`Router# issu commitversion`
		Upgrades all subpackages, including the RPBase subpackage, which is the last subpackage that needs to be upgraded.
		
		Note This step is required to ensure that all subpackages on the router were upgraded as part of this procedure, and might upgrade some subpackages that would otherwise be missed in the process.

Step 19	**show version installed** **Example:** `Router# show version installed`	(Optional) Verify that the subpackages are properly installed.
Step 20	**reload** **Example:** `Router# reload`	Reload the RP. **Tip** The router will continue normal operation even without a reload, so you can reload the router during scheduled maintenance or a slower traffic period.
Step 21	**request platform software package clean**	(Optional) Removes the unused subpackages from the router.

Examples

The following example shows the software upgrade for Cisco ASR 1002 or 1004 Router running sub-packages:

```
Router# show version
Cisco IOS Software, IOS-XE Software (X86_64_LINUX_IOSD-ADVENTERPRISEK9-M), Version
15.3(2)S, RELEASE SOFTWARE (fc1)

<output removed for brevity>

System image file is "bootflash:Active_Dir/packages.conf"

<output removed for brevity>

Router# show platform
Chassis type: ASR1004

Slot      Type                State                 Insert time (ago)
--------- ------------------- --------------------- -----------------
 0         ASR1000-2T+20X1GE   ok                     00:04:19
 0/0       BUILT-IN-2T+20X1GE  ok                     00:02:36
 1         ASR1000-SIP10       ok                     00:04:19
 1/0       SPA-2X1GE-V2        ok                     00:03:07
 1/1       SPA-10X1GE-V2       ok                     00:03:00
```

```
 R0        ASR1000-RP2          ok                    00:04:19
 R0/0                           ok, active            00:04:19
 R0/1                           ok, standby           00:02:41
 F0        ASR1000-ESP40        ok, active            00:04:19
 P0        ASR1004-PWR-AC       ok                    00:03:28
 P1        ASR1004-PWR-AC       ps, fail              00:03:28


Slot      CPLD Version        Firmware Version
--------- ------------------- ------------------------------------
 0         00200800            15.2(1r)S
 1         07091401            15.3(3r)S
 R0        08103002            15.3(3r)S
 F0        1003190E            15.3(3r)S

Router# show version installed
Package: Provisioning File, version: n/a, status: active
  File: bootflash:Active_Dir/packages.conf, on: RP0
  Built: n/a, by: n/a

  File SHA1 checksum: a624f70f68c60292f4482433f43afd92487a55c4

Package: rpbase, version: 03.12.01.S.154-2.S,
status: active

 File: bootflash:Active_Dir/asr1000rp2-rpbase.03.12.01.S.154-2.S.pkg,
on: RP0 Built: 2013-03-25_18.48, by: mcpre

File SHA1 checksum: 3a9675142898cfac350d4e42f0e37bd9f4e48538
Package: rpcontrol, version: 03.12.01.S.154-2.S,
status: active

File: bootflash:Active_Dir/asr1000rp2-rpcontrol.03.12.01.S.154-2.S.pkg,
on: RP0/0 Built: 2013-03-25_18.48, by: mcpre

File SHA1 checksum: 87b11f863f67fdf2610ee0769b929baab4c3efad

<output removed for brevity>


Router#show redundancy states
my state = 13 -ACTIVE
peer state = 8  -STANDBY HOT
Mode = Duplex
Unit = Primary
Unit ID = 48
Redundancy Mode (Operational) = sso
Redundancy Mode (Configured)  = sso
Redundancy State              = sso
Maintenance Mode = Disabled
Manual Swact = enabled
Communications = Up
client count = 107
client_notification_TMR = 30000 milliseconds
RF debug mask = 0x0


Router# mkdir bootflash:harddisk:Target_Subs
Create directory filename [Target_Subs]?
Created dir harddisk:/Target_SubsRouter

Router(config)# ip tftp source-interface gigabitethernet 0

Router# copy tftp: harddisk:Target_Subs
```

```
 Address or name of remote host []? 202.153.144.25
Source filename []?
/auto/tftp-srg-india/asr1000rp2-adventerprisek9.03.13.00.S.154-3.S-ext.bin
destination filename [/Target_Subs/asr1000rp2-adventerprisek9.03.13.00.S.154-3.S-ext.bin]?
accessing
tftp://202.153.144.25//auto/tftp-srg-india/asr1000rp2-adventerprisek9.03.13.00.S.154-3.S-e
xt.bin...
loading /auto/tftp-srg-india/asr1000rp2-adventerprisek9.03.13.00.S.154-3.S-ext.bin from
202.153.144.25 (via GigabitEthernet0): !!!!!!!!!!!!!!!!O!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
O!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!O!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!
[OK - 569597380 bytes]
569597380 bytes copied in 101.618 secs
(5605280 bytes/sec)

Router# request platform software package expand file
harddisk:/Target_Subs/asr1000rp2-adventerprisek9.03.13.00.S.154-3.S-ext.bin

Verifying parameters
Validating package type
Copying package files
SUCCESS: Finished expanding all-in-one software package.

Router# dir harddisk:Target_Subs
Directory of harddisk:/Target_Subs/

9666562  -rw-   569597380   Aug 3 2013 13:25:06 +05:30
asr1000rp2-adventerprisek9.03.13.00.S.154-3.S-ext.bin
666566  -rw-   37557200   Aug 3 2013 13:27:06 +05:30
asr1000rp2-elcbase.03.13.00.S.154-3.S-ext.pkg
666567  -rw-   51194832   Aug 3 2013 13:27:07 +05:30
asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg
666568  -rw-   80657364   Aug 3 2013 13:27:07 +05:30
asr1000rp2-espbase.03.13.00.S.154-3.S-ext.pkg
666569  -rw-   95446456   Aug 3 2013 13:27:08 +05:30
asr1000rp2-espx86base.03.13.00.S.154-3.S-ext.pkg
666564  -rw-   95449381   Aug 3 2013 13:27:06 +05:30
asr1000rp2-packages-adventerprisek9.03.13.00.S.154-3.S-ext.conf
666570  -rw-    23350232   Aug 3 2013 13:27:08 +05:30
asr1000rp2-rpaccess.03.13.00.S.154-3.S-ext.pkg
666571  -rw-    37694900   Aug 3 2013 13:27:08 +05:30
asr1000rp2-rpbase.03.13.00.S.154-3.S-ext.pkg
666572  -rw-    45536216   Aug 3 2013 13:27:08 +05:30
asr1000rp2-rpcontrol.03.13.00.S.154-3.S-ext.pkg
666573  -rw-   118754284   Aug 3 2013 13:27:08 +05:30
asr1000rp2-rpios-adventerprisek9.03.13.00.S.154-3.S-ext.pkg
666574  -rw-    38380500   Aug 3 2013 13:27:08 +05:30
asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg
666575  -rw-    61760468   Aug 3 2013 13:27:08 +05:30
asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg
666565  -rw-       10165   Aug 3 2013 13:27:08 +05:30  packages.conf


78704144384 bytes total (47541493760 bytes free)


Router# copy harddisk:Target_Subs/asr1000rp2-espbase.03.13.00.S.154-3.S-ext.pkg bootflash:
Active_Dir/
Destination filename [Active_Dir/asr1000rp2-espbase.03.13.00.S.154-3.S-ext.pkg]?
copy in
progress...CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
```

```
CCCC
```

```
Active_Dir/Destination filename
[Active_Dir/asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg]?
copy in
progress...CCCC
```

```
Verifying existing software set
Processing candidate provisioning file
Constructing working set for candidate package set
Constructing working set for running package set
Checking command output
Constructing merge of running and candidate packages
Checking if resulting candidate package set would be complete
Finished candidate package set construction

--- Starting compatibility testing ---
Determining whether candidate package set is compatible
WARNING:Candidate software combination not found in compatibility database
WARNING:Determining whether installation is valid
Creating matrix_file by locate_latest_matrix_file  /tmp/issu/provision/s
WARNING: Candidate software combination not found in compatibility database
WARNING: Candidate software combination not found in compatibility database
WARNING:Software sets are identified as compatible
Verifying image type compatibility
Checking IPC compatibility with running software
Checking candidate package set infrastructure compatibility
Checking infrastructure compatibility with running software
Checking package specific compatibility
Finished compatibility testing
--- Starting impact testing ---
Checking operational impact of change
WARNING: Connection may be lost during installation of IOS package
Finished impact testing
--- Starting list of software package changes ---
No old package files removed
New files list:
  Added asr1000rp2-rpaccess.03.13.00.S.154-3.S-ext.pkg
  Added asr1000rp2-rpcontrol.03.13.00.S.154-3.S-ext.pkg
  Added asr1000rp2-rpios-adventerprisek9.03.13.00.S.154-3.S-ext.pkg
Finished list of software package changes
--- Starting commit of software changes ---
Updating provisioning rollback files
Creating pending provisioning file
Committing provisioning file

Finished commit of software changes
--- Starting analysis of software changes ---

Finished analysis of software changes
--- Starting update running software ---
Blocking peer synchronization of operating information
Creating the command set placeholder directory
 Finding latest command set
 Finding latest command shortlist lookup file
 Finding latest command shortlist file
 Assembling CLI output libraries
 Assembling CLI input libraries
 Assembling Dynamic configuration files
 Applying interim IPC and database definitions
 Replacing running software
 Replacing CLI software
 Restarting software
 Restarting IOS PID: 21552, in slot/bay 0/1
 Applying final IPC and database definitions
 Generating software version information
 Notifying running software of updates
 Unblocking peer synchronization of operating information
 Unmounting old packages
 Cleaning temporary installation files
 Finished update running software.
```

```
SUCCESS: Finished installing software.



Router#issu commitversion
-- Starting local lock acquisition on R0----
Finished local lock acquisition on R0
-- Starting installation changes ---
Cancelling rollback timer
Finished installation changes

SUCCESS: Installation changes committed

Router#redundancy force-switchover
Proceed with switchover to standby RP? [confirm]
Manual Swact = enabled

%IOSXE_INFRA-6-CONSOLE_ACTIVE: R0/1 console active.  Press RETURN to get started![OK]
*Aug  3 13:43:52.101 IST: %CMANRP-6-CMHASTATUS: RP switchover, received chassis event to
become active
*Aug  3 13:43:52.193 IST: %REDUNDANCY-3-SWITCHOVER: RP switchover (PEER_NOT_PRESENT)
*Aug  3 13:43:52.194 IST: %REDUNDANCY-3-SWITCHOVER: RP switchover (PEER_DOWN)
*Aug  3 13:43:52.194 IST: %REDUNDANCY-3-SWITCHOVER: RP switchover
(PEER_REDUNDANCY_STATE_CHANGE)
*Aug  3 13:43:51.231 IST: %RF-5-RF_RELOAD: Self reload. Reason: redundancy
force-switchover
*Aug  3 13:43:51.461 IST: %SYS-5-SWITCHOVER: Switchover requested by
red_switchover_process. Reason: redundancy force-switchover.
*Aug  3 13:43:52.198 IST: %PLATFORM-6-HASTATUS: RP switchover, sent message became active.
IOS is ready to switch to primary after chassis confirmation
*Aug  3 13:43:52.200 IST: %CMANRP-6-CMHASTATUS: RP switchover, received chassis event
became active
*Aug  3 13:43:52.449 IST: %PLATFORM-6-HASTATUS_DETAIL: RP switchover, received chassis
event became active. Switch to primary (count 1)
*Aug  3 13:43:52.733 IST: %LINEPROTO-5-UPDOWN: Line protocol on Interface
GigabitEthernet0, changed state to down
*Aug  3 13:43:53.098 IST: % Redundancy mode change to SSO
*Aug  3 13:43:53.126 IST: %LINK-3-UPDOWN: Interface Lsmpi0, changed state to up
*Aug  3 13:43:53.127 IST: %LINK-3-UPDOWN: Interface EOBC0, changed state to up
*Aug  3 13:43:53.127 IST: %LINK-3-UPDOWN: Interface LIIN0, changed state to up
*Aug  3 13:43:54.127 IST: %LINEPROTO-5-UPDOWN: Line protocol on Interface Lsmpi0, changed
state to up
*Aug  3 13:43:54.127 IST: %LINEPROTO-5-UPDOWN: Line protocol on Interface EOBC0, changed
state to up
*Aug  3 13:43:54.127 IST: %LINEPROTO-5-UPDOWN: Line protocol on Interface LIIN0, changed
state to up
*Aug  3 13:43:55.117 IST: %LINK-3-UPDOWN: Interface Null0, changed state to up
*Aug  3 13:43:55.117 IST: %LINK-3-UPDOWN: Interface GigabitEthernet0/0/8, changed state to
up
*Aug  3 13:43:55.117 IST: %LINK-3-UPDOWN: Interface GigabitEthernet0/0/10, changed state
to up
*Aug  3 13:43:55.117 IST: %LINK-3-UPDOWN: Interface GigabitEthernet1/0/0, changed state to
up
*Aug  3 13:43:55.119 IST: %LINK-3-UPDOWN: Interface GigabitEthernet1/0/1, changed state to
up
*Aug  3 13:43:55.119 IST: %LINK-3-UPDOWN: Interface GigabitEthernet1/1/0, changed state to
up
*Aug  3 13:43:55.119 IST: %LINK-3-UPDOWN: Interface GigabitEthernet1/1/1, changed state to
up
*Aug  3 13:43:55.119 IST: %LINK-3-UPDOWN: Interface GigabitEthernet1/1/3, changed state to
up
```

```
*Aug  3 13:43:55.119 IST: %LINK-3-UPDOWN: Interface GigabitEthernet1/1/5, changed state to
up
*Aug  3 13:43:55.124 IST: %LINK-5-CHANGED: Interface GigabitEthernet0/0/0, changed state
to administratively down
*Aug  3 13:43:55.124 IST: %LINK-5-CHANGED: Interface GigabitEthernet0/0/1, changed state
to administratively down
*Aug  3 13:43:55.124 IST: %LINK-5-CHANGED: Interface GigabitEthernet0/0/2, changed state
to administratively down
*Aug  3 13:43:55.124 IST: %LINK-5-CHANGED: Interface GigabitEthernet0/0/3, changed state
to administratively down
*Aug  3 13:43:55.124 IST: %LINK-5-CHANGED: Interface GigabitEthernet0/0/4, changed state
to administratively down
*Aug  3 13:43:55.124 IST: %LINK-5-CHANGED: Interface GigabitEthernet0/0/5, changed state
to administratively down
*Aug  3 13:43:55.124 IST: %LINK-5-CHANGED: Interface GigabitEthernet0/0/6, changed state
to administratively down
*Aug  3 13:43:55.124 IST: %LINK-5-CHANGED: Interface GigabitEthernet0/0/7, changed state
to administratively down
*Aug  3 13:43:55.124 IST: %LINK-5-CHANGED: Interface GigabitEthernet0/0/9, changed state
to administratively down
*Aug  3 13:43:55.124 IST: %LINK-5-CHANGED: Interface GigabitEthernet0/0/11, changed state
to administratively down
*Aug  3 13:43:56.118 IST: %LINEPROTO-5-UPDOWN: Line protocol on Interface Null0, changed
state to up
*Aug  3 13:43:56.118 IST: %LINEPROTO-5-UPDOWN: Line protocol on Interface
GigabitEthernet0/0/8, changed state to up
*Aug  3 13:43:56.118 IST: %LINEPROTO-5-UPDOWN: Line protocol on Interface
GigabitEthernet0/0/10, changed state to up
*Aug  3 13:43:56.118 IST: %LINEPROTO-5-UPDOWN: Line protocol on Interface
TenGigabitEthernet0/0/20, changed state to down
*Aug  3 13:43:56.118 IST: %LINEPROTO-5-UPDOWN: Line protocol on Interface
TenGigabitEthernet0/0/21, hanged state to down
*Aug  3 13:43:56.118 IST: %LINEPROTO-5-UPDOWN: Line protocol on Interface
GigabitEthernet1/0/0, changed state to up
*Aug  3 13:43:56.120 IST: %LINEPROTO-5-UPDOWN: Line protocol on Interface
GigabitEthernet1/0/1, changed state to up
*Aug  3 13:43:56.120 IST: %LINEPROTO-5-UPDOWN: Line protocol on Interface
GigabitEthernet1/1/0, changed state to up
*Aug  3 13:43:56.120 IST: %LINEPROTO-5-UPDOWN: Line protocol on Interface
GigabitEthernet1/1/1, changed state to up
*Aug  3 13:43:56.120 IST: %LINEPROTO-5-UPDOWN: Line protocol on Interface
GigabitEthernet1/1/3, changed state to up
*Aug  3 13:44:05.127 IST: %LINK-3-UPDOWN: Interface GigabitEthernet0, changed state to up
*Aug  3 13:44:06.127 IST: %LINEPROTO-5-UPDOWN: Line protocol on Interface
GigabitEthernet0, changed state to up
*Aug  3 13:44:32.069 IST: %REDUNDANCY-5-PEER_MONITOR_EVENT: Active detected a standby
insertion (raw-event=PEER_FOUND(4))
*Aug  3 13:44:32.070 IST: %REDUNDANCY-5-PEER_MONITOR_EVENT: Active detected a standby
insertion (raw-event=PEER_REDUNDANCY_STATE_CHANGE(5))
*Aug  3 13:44:34.969 IST: %REDUNDANCY-3-IPC: IOS versions do not match.
*Aug  3 13:45:40.946 IST: %RF-5-RF_TERMINAL_STATE: Terminal state reached for (SSO)


Router# issu loadversion rp 0 file
bootflash:Active_Dir/asr1000rp2-{rpaccess,rpios,rpcontrol}*03.13.00.S.154-3.S-ext*.pkg bay
0 force
--- Starting local lock acquisition on R0 ---
Finished local lock acquisition on R0
--- Starting file path checking ---
Finished file path checking
---Starting image file verification---
Checking image file names
Locating image files and validating name syntax
 Found asr1000rp2-rpaccess.03.13.00.S.154-3.S-ext.pkg
```

```
 Found asr1000rp2-rpios-adventerprisek9.03.13.00.S.154-3.S-ext.pkg
 Found asr1000rp2-rpcontrol.03.13.00.S.154-3.S-ext.pkg
Verifying image file locations
Inspecting image file types
Processing image file constraints
Creating candidate provisioning file
Finished image file verification
--- Starting candidate package set construction ---
Verifying existing software set
Processing candidate provisioning file
Constructing working set for candidate package set
Constructing working set for running package set
Checking command outpu
Constructing merge of running and candidate packages
Checking if resulting candidate package set would be complete
Finished candidate package set construction
--- Starting compatibility testing ---
Determining whether candidate package set is compatible
WARNING: Candidate software combination not found in compatibility database
WARNING:Determining whether installation is valid
Creating matrix_file by locate_latest_matrix_file/tmp/issu/provision/sw
Software sets are identified as compatible
Verifying image type compatibility
Checking IPC compatibility with running software
Checking candidate package set infrastructure compatibility
Checking infrastructure compatibility with running software
Checking package specific compatibility
Finished compatibility testing

--- Starting impact testing ---
Checking operational impact of change

WARNING: Connection may be lost during installation of IOS package
Finished impact testing
--- Starting list of software package changes ---
Old files list:
  Removed asr1000rp2-rpaccess.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-rpcontrol.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-rpios-adventerprisek9.03.12.01.S.154-2.S.pkg
No new package files added
  Finished list of software package changes
--- Starting commit of software changes ---
Updating provisioning rollback files
Creating pending provisioning file
Committing provisioning file
Finished commit of software changes
--- Starting analysis of software changes ---
Finished analysis of software changes
Starting update running software
Blocking peer synchronization of operating information
Creating the command set placeholder directory
 Finding latest command set
 Finding latest command shortlist lookup file
 Finding latest command shortlist file
 Assembling CLI output libraries
 Assembling CLI input libraries
 Assembling Dynamic configuration files
 Applying interim IPC and database definitions
 Replacing running software
 Replacing CLI software
  Restarting software
  Restarting IOS PID: 25452, in slot/bay 0/0
*Aug  3 13:48:07.051 IST:
%REDUNDANCY-3-STANDBY_LOST: Standby processor fault (PEER_NOT_PRESENT)
```

```
*Aug  3 13:48:07.052 IST: %REDUNDANCY-3-STANDBY_LOST: Standby processor fault (PEER_DOWN)
*Aug  3 13:48:07.052 IST: %REDUNDANCY-3-STANDBY_LOST: Standby processor fault
(PEER_REDUNDANCY_STATE_CHANGE)
*Aug  3 13:48:09.683 IST: %RF-5-RF_RELOAD: Peer reload.
Reason: EHSA standby down
*Aug  3 13:48:09.692 IST: % Redundancy mode change to SSO
*Aug  3 13:48:49.450 IST: %REDUNDANCY-5-PEER_MONITOR_EVENT: Active detected a standby
insertion (raw-event=PEER_FOUND(4))
*Aug  3 13:48:49.451 IST: %REDUNDANCY-5-PEER_MONITOR_EVENT: Active detected a standby
insertion (raw-event=PEER_REDUNDANCY_STATE_CHANGE(5))
*Aug  3 13:48:50.035 IST:
%ISSU-3-PEER_IMAGE_NOT_IN_INCOMP_LIST: Peer image (X86_64_LINUX_IOSD-ADVENTERPRISEK9-M),
version (15.3(3)S) on peer uid (48) is not in the incompatible images list
*Aug  3 13:48:50.036 IST:
%ISSU-3-PEER_IMAGE_NOT_IN_INCOMP_LIST: Peer image (X86_64_LINUX_IOSD-ADVENTERPRISEK9-M),
version (15.3(3)S) on peer uid (48) is not in the incompatible images list
Applying final IPC and database definitions
*Aug  3 13:48:53.521 IST: %DYNCMD-7-CMDSET_UNLOADED:
The Dynamic Command set from the Shell Manager has been unloaded
Generating software version information
  Notifying running software of updates
  Unblocking peer synchronization of operating information
  Unmounting old packages
  Cleaning temporary installation files
  Finished update running software

SUCCESS: Finished installing software.

Router#
*Aug  3 13:49:57.477 IST: %NBAR_HA-5-NBAR_INFO: NBAR sync DONE!
*Aug  3 13:49:57.893 IST: %ISSU-3-PEER_IMAGE_REM_FROM_INCOMP_LIST: Peer image
(X86_64_LINUX_IOSD-ADVENTERPRISEK9-M),
version (15.3(2)S) on peer uid (48) being removed from the incompatibility list
*Aug  3 13:49:57.893 IST: %HA_CONFIG_SYNC-6-BULK_CFGSYNC_SUCCEED: Bulk Sync succeeded
*Aug  3 13:49:57.895 IST: %RF-5-RF_TERMINAL_STATE: Terminal state reached for (SSO)


Router# issu commitversion
--- Starting local lock acquisition on R0--
Finished local lock acquisition on R0
Starting installation changes
Cancelling rollback timer
Finished installation changes
SUCCESS: Installation changes committed


Router# issu loadversion rp 0 file
bootflash:Active_Dir/asr1000rp2-{sipbase,sipspa}*03.13.00.S.154-3.S-ext*.pkg slot 1 force

--- Starting local lock acquisition on R0 ---
Finished local lock acquisition on R0
--- Starting file path checking ---
Finished file path checking
--- Starting image file verification ---
Checking image file names
Locating image files and validating name syntax
  Found asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg
Verifying image file locations
Inspecting image file types
Processing image file constraints
Creating candidate provisioning file
Finished image file verification
--- Starting candidate package set construction ---
```

```
Verifying existing software set
Processing candidate provisioning file
Constructing working set for candidate package set
Constructing working set for running package set
Checking command output
Constructing merge of running and candidate packages
Checking if resulting candidate package set would be complete
Finished candidate package set construction
--- Starting compatibility testing ---
Determining whether candidate package set is compatible
WARNING: Candidate software combination not found in compatibility database
WARNING:Determining whether installation is valid
Creating matrix_file by locate_latest_matrix_file  /tmp/issu/provision/s
WARNING: Candidate software combination not found in compatibility database
WARNING: Candidate software combination not found in compatibility database
WARNING: Software sets are identified as compatible

Verifying image type compatibility
Checking IPC compatibility with running software
Checking candidate package set infrastructure compatibility
Checking infrastructure compatibility with running software
Checking package specific compatibility
Finished compatibility testing
--- Starting impact testing-----
Checking operational impact of change
Finished impact testing
--- Starting list of software package changes---
No old package files removed
New files list:
  Added asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg
  Added asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg
Finished list of software package changes
--- Starting commit of software changes ---
Updating provisioning rollback files
Creating pending provisioning file
Committing provisioning file
Finished commit of software changes
--- Starting analysis of software changes ---
Finished analysis of software changes
--- Starting update running software ---
Blocking peer synchronization of operating information
Creating the command set placeholder directory
  Finding latest command set
  Finding latest command shortlist lookup file
  Finding latest command shortlist file
  Assembling CLI output libraries
  Assembling CLI input libraries
  Assembling Dynamic configuration files
  Applying interim IPC and database definitions
  Replacing running software
  Replacing CLI software
  Restarting software
  Restarting SIP1
  Applying final IPC and database definitions
*Aug  3 13:52:05.767 IST: %IOSXE_OIR-6-OFFLINECARD: Card (cc) offline in slot 1
*Aug  3 13:52:05.770 IST: %IOSXE_OIR-6-REMSPA: SPA removed from subslot 1/0, interfaces
disabled
*Aug  3 13:52:05.770 IST: %IOSXE_OIR-6-REMSPA: SPA removed from subslot 1/1, interfaces
disabled
*Aug  3 13:52:05.778 IST: %SPA_OIR-6-OFFLINECARD: SPA (SPA-2X1GE-V2) offline in subslot
1/0
*Aug  3 13:52:05.786 IST: %SPA_OIR-6-OFFLINECARD: SPA (SPA-10X1GE-V2) offline in subslot
1/1
Generating software version information
```

```
Notifying running software of updates
Unblocking peer synchronization of operating information
Unmounting old packages
Cleaning temporary installation files
  Finished update running software

SUCCESS: Finished installing software.
Router#
*Aug  3 13:53:00.262 IST: %IOSXE_OIR-6-ONLINECARD: Card (cc) online in slot 1
*Aug  3 13:53:00.901 IST: %IOSXE_OIR-6-INSSPA: SPA inserted in subslot 1/0
*Aug  3 13:53:01.194 IST: %IOSXE_OIR-6-INSSPA: SPA inserted in subslot 1/1
*Aug  3 13:53:18.148 IST: %LINK-3-UPDOWN: SIP1/0: Interface EOBC1/1, changed state to up
*Aug  3 13:53:23.978 IST: %SPA_OIR-6-ONLINECARD: SPA (SPA-2X1GE-V2) online in subslot 1/0
*Aug  3 13:53:23.609 IST: %LINK-3-UPDOWN: SIP1/1: Interface EOBC1/1, changed state to up
*Aug  3 13:53:26.288 IST: %LINK-3-UPDOWN: Interface GigabitEthernet1/0/0, changed state to
up
*Aug  3 13:53:26.471 IST: %LINK-3-UPDOWN: Interface GigabitEthernet1/0/1, changed state to
up
*Aug  3 13:53:26.605 IST: %LINK-3-UPDOWN: SIP1/0: Interface GigabitEthernet1/0/0, changed
state to up
*Aug  3 13:53:29.056 IST: %SPA_OIR-6-ONLINECARD: SPA (SPA-10X1GE-V2) online in subslot 1/1
*Aug  3 13:53:31.020 IST: %LINK-3-UPDOWN: Interface GigabitEthernet1/1/1, changed state to
up
*Aug  3 13:53:31.030 IST: %LINK-3-UPDOWN: Interface GigabitEthernet1/1/3, changed state to
up
*Aug  3 13:53:31.042 IST: %LINK-3-UPDOWN: Interface GigabitEthernet1/1/4, changed state to
down
*Aug  3 13:53:31.045 IST: %LINK-3-UPDOWN: Interface GigabitEthernet1/1/5, changed state to
up
*Aug  3 13:53:31.046 IST: %LINK-3-UPDOWN: Interface GigabitEthernet1/1/6, changed state to
down
*Aug  3 13:53:31.048 IST: %LINK-3-UPDOWN: Interface GigabitEthernet1/1/7, changed state to
down
*Aug  3 13:53:31.051 IST: %LINK-3-UPDOWN: Interface GigabitEthernet1/1/8, changed state to
down
*Aug  3 13:53:31.053 IST: %LINK-3-UPDOWN: Interface GigabitEthernet1/1/9, changed state to
down
*Aug  3 13:53:31.312 IST: %LINK-3-UPDOWN: Interface GigabitEthernet1/1/0, changed state to
up
*Aug  3 13:53:32.833 IST: %LINK-3-UPDOWN: SIP1/1: Interface GigabitEthernet1/1/0, changed
state to up
*Aug  3 13:53:33.007 IST: %LINK-3-UPDOWN: SIP1/1: Interface GigabitEthernet1/1/7, changed
state to down


Router# issu commitversion
--- Starting local lock acquisition on R0 ---
Finished local lock acquisition on R0
Starting installation changes
Cancelling rollback timer
Finished installation changes

SUCCESS: Installation changes committed


Router# issu loadversion rp 0 file
bootflash:Active_Dir/asr1000rp2-{elcbase,elcspa}*03.13.00.S.154-3.S-ext*.pkg slot 0 force
Starting local lock acquisition on R0
---Finished local lock acquisition on R0---
Starting file path checking
---Finished file path checking---
Starting image file verification
---Checking image file names----
Locating image files and validating name syntax
```

```
  Found asr1000rp2-elcbase.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg
Verifying image file locations
Inspecting image file types
Processing image file constraints
Creating candidate provisioning file
Finished image file verification
--- Starting candidate package set construction ---
Verifying existing software set
Processing candidate provisioning file
Constructing working set for candidate package set
Constructing working set for running package set
Checking command output
Constructing merge of running and candidate packages
Checking if resulting candidate package set would be complete
Finished candidate package set construction
--- Starting compatibility testing ---
Determining whether candidate package set is compatible

WARNING: Candidate software combination not found in compatibility database
WARNING: Determining whether installation is valid
Creating matrix_file by locate_latest_matrix_file  /tmp/issu/provision/s
WARNING: Candidate software combination not found in compatibility database
WARNING: Candidate software combination not found in compatibility database
WARNING: Software sets are identified as compatible
Verifying image type compatibility
Checking IPC compatibility with running software
Checking candidate package set infrastructure compatibility
Checking infrastructure compatibility with running software
Checking package specific compatibility
Finished compatibility testing
--- Starting impact testing---
Checking operational impact of change
Finished impact testing
--- Starting list of software package changes ---
No old package files removed
New files list:
  Added asr1000rp2-elcbase.03.13.00.S.154-3.S-ext.pkg
  Added asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg
Finished list of software package changes

--- Starting commit of software changes ---
Updating provisioning rollback files
Creating pending provisioning file
Committing provisioning file
Finished commit of software changes
--- Starting analysis of software changes ---
Finished analysis of software changes
--- Starting update running software ---
Blocking peer synchronization of operating information
Creating the command set placeholder directory
  Finding latest command set
  Finding latest command shortlist lookup file
  Finding latest command shortlist file
  Assembling CLI output libraries
  Assembling CLI input libraries
  Assembling Dynamic configuration files
  Applying interim IPC and database definitions
  Replacing running software
  Replacing CLI software
  Restarting software
  Applying final IPC and database definitions
  Generating software version information
  Notifying running software of updates
```

```
  Unblocking peer synchronization of operating information
  Unmounting old packages
  Cleaning temporary installation files
  Finished update running software

SUCCESS: Finished installing software.

Router#issu commitversion
-- Starting local lock acquisition on R0 ---
Finished local lock acquisition on R0
--- Starting installation changes ---
Cancelling rollback timer
Finished installation changes

SUCCESS: Installation changes committed


Router# issu loadversion rp 0 file
bootflash:Active_Dir/asr1000rp2-esp*03.13.00.S.154-3.S-ext*.pkg force
--- Starting local lock acquisition on R0 ---
Finished local lock acquisition on R0
--- Starting file path checking ---
Finished file path checking
--- Starting image file verification ---
Checking image file names
Locating image files and validating name syntax
  Found asr1000rp2-espbase.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-espx86base.03.13.00.S.154-3.S-ext.pkg
Verifying image file locations
Inspecting image file types
Processing image  file constraints
Creating candidate provisioning file
Finished image file verification
--- Starting candidate package set construction --
Verifying existing software set
Processing candidate provisioning file
Constructing working set for candidate package set
Constructing working set for running package set
Checking command output
Constructing merge of rnning and candidate packages
Checking if resulting candidate package set would be complete
Finished candidate package set construction

--- Starting compatibility testing ---
Determining whether candidate package set is compatible
WARNING: Candidate software combination not found in compatibility database
WARNING: Determining whether installation is valid
Creating matrix_file by locate_latest_matrix_file  /tmp/issu/provision/s
WARNING: Candidate software combination not found in compatibility database
WARNING: Candidate software combination not found in compatibility database
WARNING: Software sets are identified as compatible
Verifying image type compatibility
Checking IPC compatibility with running software
Checking candidate package set infrastructure compatibility
Checking infrastructure compatibility with running software
Checking package specific compatibility
Finished compatibility testing
--- Starting impact testing ---
Checking operational impact of change
Finished  impact testing
--- Starting list of software package changes ---
Old files list:
  Removed asr1000rp2-espbase.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2- espx86base.03.12.01.S.154-2.S.pkg
```

```
New files list:
  Added asr1000rp2-espbase.03.13.00.S.154-3.S-ext.pkg
  Added asr1000rp2-espx86base.03.13.00.S.154-3.S-ext.pkg
Finished list of software package changes
--- Starting commit of software changes ---
Updating provisioning rollback files
Creating pending provisioning file
Committing provisioning file
Finished commit of software changes
Starting analysis of software changes
Finished analysis of software changes---
Starting update running software
Blocking peer synchronization of operating information
Creating the command set placeholder directory
  Finding latest command set
  Finding latest command shortlist lookup file
  Finding latest command shortlist file
  Assembling CLI output libraries
  Assembling CLI input libraries
  Assembling Dynamic configuration files
  Applying interim IPC and database definitions
  Replacing running software
  Replacing CLI software
  Restarting software
  Restarting ESP0
  Applying final IPC and database definitions
*Aug  3 14:02:51.450 IST: %IOSXE_OIR-6-OFFLINECARD: Card (fp) offline in slot F0
Generating software version information
  Notifying running software of updates
  Unblocking peer synchronization of operating information
  Unmounting old packages
  Cleaning temporary installation files
  Finished update running software

SUCCESS: Finished installing software.

Router#
*Aug  3 14:04:49.802 IST: %CPPHA-7-START: F0: cpp_ha:  CPP 0
preparing image /tmp/sw/fp/0/0/fpx86/mount/usr/cpp/bin/qfp-ucode-esp40
*Aug  3 14:04:50.172 IST: %CPPHA-7-START: F0: cpp_ha:  CPP 0 startup init image
/tmp/sw/fp/0/0/fpx86/mount/usr/cpp/bin/qfp-ucode-esp40
*Aug  3 14:04:50.746 IST: %IOSXE_OIR-6-ONLINECARD: Card (fp) online in slot F0
*Aug  3 14:04:55.480 IST: %CPPHA-7-START: F0: cpp_ha:  CPP 0 running init image
/tmp/sw/fp/0/0/fpx86/mount/usr/cpp/bin/qfp-ucode-esp40
*Aug  3 14:04:55.698 IST: %CPPHA-7-READY: F0: cpp_ha:  CPP 0 loading and initialization
complete
*Aug  3 14:04:55.837 IST: %IOSXE-6-PLATFORM: F0: cpp_cp: Process
CPP_PFILTER_EA_EVENT__API_CALL__REGISTER


Router#issu commitversion
-- Starting local lock acquisition on R0 ---
Finished local lock acquisition on R0
---Starting installation changes ---
Cancelling rollback timer
Finished installation changes

SUCCESS: Installation changes committed


Router# issu loadversion rp 0 file
bootflash:Active_Dir/asr1000rp*03.13.00.S.154-3.S-ext*.pkg
-- Starting local lock acquisition on R0
---Finished local lock acquisition on R0
```

```
Starting file path checking
Finished file path checking
Starting image file verification
Checking image file names
Locating image files and validating name syntax
  Found asr1000rp2-elcbase.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-espbase.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-espx86base.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-rpaccess.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-rpbase.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-rpcontrol.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-rpios-adventerprisek9.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg
Verifying image file locations
Inspecting image file types
    WARNING: In-service installation of RP Base package
    WARNING: requires software reboot of target RP
    WARNING: Automatically setting the on-reboot flag
Processing image file constraints
Creating candidate provisioning file
Finished image file verification
--- Starting candidate package set construction ---
Verifying existing software set
Processing candidate provisioning file
Constructing working set for candidate package set
Constructing working set for running package set
Checking command output
Constructing merge of running and candidate packages
Checking if resulting candidate package set would be complete
Finished candidate package set construction
--- Starting compatibility testing ---

Determining whether candidate package set is compatible
Determining whether installation is valid
Determining whether installation is valid ... skipped
Verifying image type compatibility
Checking IPC compatibility for candidate software
Checking  candidate package set infrastructure compatibility
Checking infrastructure compatibility with running software
Checking infrastructure compatibility with running software ... skipped
Checking package specific compatibility
Finished compatibility testing
--- Starting list of software package changes ---
Old files list:
  Removed asr1000rp2-elcbase.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-elcspa.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-rpbase.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-sipbase.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-sipspa.03.12.01.S.154-2.S.pkg

New files list:
  Added asr1000rp2-rpbase.03.13.00.S.154-3.S-ext.pkg
Finished list of software package changes
--- Starting commit of software changes ---
Updating provisioning rollback files
Creating pending provisioning file
Committing provisioning file
Finished commit of software changes

SUCCESS: Software provisioned.  New software will load on reboot.
```

```
Router# show version r0 provisioned
Package: Provisioning File, version: n/a, status: active
File: bootflash:Active_Dir/packages.conf,
on: RP0 Built: n/a, by:n/a
File SHA1 checksum: c79075780592aec1312725f4a2357a034fda2d3b

Package: rpbase,
version: 03.13.00.S.154-3.S-ext, status: n/a
File: bootflash:Active_Dir/asr1000rp2-rpbase.03.13.00.S.154-3.S-ext.pkg,
on: RP0   Built: 2013-07-25_22.55, by: mcpre
File SHA1 checksum: 4f655c54bb95b4dfa24a0d25ebf97cf8527c69e9

Package: rpcontrol, version: 03.13.00.S.154-3.S-ext, status: n/a
File: bootflash:Active_Dir/asr1000rp2-rpcontrol.03.13.00.S.154-3.S-ext.pkg,
on: RP0/0   Built: 2013-07-25_22.55, by: mcpre
  File SHA1 checksum: 8a0a45ea5c7a656c0eef6726174461584f182c78

Package: rpios-adventerprisek9,version: 03.13.00.S.154-3.S-ext, status: n/a
File: bootflash:Active_Dir/asr1000rp2-rpios-adventerprisek9.03.13.00.S.154-3.S-ext.pkg,
on: RP0/0   Built: 2013-07-25_23.00, by: mcpre
File SHA1 checksum: 85e9eab826bff2194ef568a56c76453625383ad2

Package: rpaccess, version: 03.13.00.S.154-3.S-ext, status: n/a

File: bootflash:Active_Dir/asr1000rp2-rpaccess.03.13.00.S.154-3.S-ext.pkg,
on: RP0/0  Built: 2013-07-25_22.55, by: mcpre
File SHA1 checksum: a360dff0fd76a9b1ae67cda9116c97b62f25ab09

Package: rpcontrol, version: 03.13.00.S.154-3.S-ext, status: n/a
File: bootflash:Active_Dir/asr1000rp2-rpcontrol.03.13.00.S.154-3.S-ext.pkg,
on: RP0/1   Built: 2013-07-25_22.55, by: mcpre
File SHA1 checksum: 8a0a45ea5c7a656c0eef6726174461584f182c78

Package: rpios-adventerprisek9, version: 03.13.00.S.154-3.S-ext, status: n/a
File: bootflash:Active_Dir/asr1000rp2-rpios-adventerprisek9.03.13.00.S.154-3.S-ext.pkg,
on: RP0/1   Built: 2013-07-25_23.00, by: mcpre
File SHA1 checksum: 85e9eab826bff2194ef568a56c76453625383ad2

Package: rpaccess, version: 03.13.00.S.154-3.S-ext, status: n/a
File: bootflash:Active_Dir/asr1000rp2-rpaccess.03.13.00.S.154-3.S-ext.pkg,
on: RP0/1  Built: 2013-07-25_22.55, by: mcpre
File SHA1 checksum: a360dff0fd76a9b1ae67cda9116c97b62f25ab09

Package: rpbase, version: 03.13.00.S.154-3.S-ext, status: n/a
File: bootflash:Active_Dir/asr1000rp2-rpbase.03.13.00.S.154-3.S-ext.pkg,
on: RP1   Built: 2013-07-25_22.55, by: mcpre
File SHA1 checksum: 4f655c54bb95b4dfa24a0d25ebf97cf8527c69e9

Package:rpcontrol, version: 03.13.00.S.154-3.S-ext, status: n/a
File: bootflash:Active_Dir/asr1000rp2-rpcontrol.03.13.00.S.154-3.S-ext.pkg, on: RP1/0
Built: 2013-07-25_22.55, by: mcpre
File SHA1 checksum: 8a0a45ea5c7a656c0eef6726174461584f182c78

Package: rpios-adventerprisek9, version: 03.13.00.S.154-3.S-ext, status: n/a
File: bootflash:Active_Dir/asr1000rp2-rpios-adventerprisek9.03.13.00.S.154-3.S-ext.pkg,
on: RP1/0  Built: 2013-07-25_23.00, by: mcpre
File SHA1 checksum: 85e9eab826bff2194ef568a56c76453625383ad2

Package: rpaccess, version: 03.13.00.S.154-3.S-ext, status: n/a
File: bootflash:Active_Dir/asr1000rp2-rpaccess.03.13.00.S.154-3.S-ext.pkg,
on: RP1/0  Built: 2013-07-25_22.55, by: mcpre
File SHA1 checksum: a360dff0fd76a9b1ae67cda9116c97b62f25ab09

Package: rpcontrol, version: 03.13.00.S.154-3.S-ext, status: n/a
```

```
File: bootflash:Active_Dir/asr1000rp2-rpcontrol.03.13.00.S.154-3.S-ext.pkg, on: RP1/1
Built: 2013-07-25_22.55, by: mcpre
File SHA1 checksum: 8a0a45ea5c7a656c0eef6726174461584f182c78

Package: rpios-adventerprisek9, version: 03.13.00.S.154-3.S-ext, status:n/a
File: bootflash:Active_Dir/asr1000rp2-rpios-adventerprisek9.03.13.00.S.154-3.S-ext.pkg,
on: RP1/1 Built: 2013-07-25_23.00, by: mcpre
File SHA1 checksum: 85e9eab826bff2194ef568a56c76453625383ad2

Package: rpaccess, version: 03.13.00.S.154-3.S-ext, status: n/a
File: bootflash:Active_Dir/asr1000rp2-rpaccess.03.13.00.S.154-3.S-ext.pkg, on: RP1/1
Built: 2013-07-25_22.55, by: mcpre
File SHA1 checksum: a360dff0fd76a9b1ae67cda9116c97b62f25ab09

Package:espbase, version: 03.13.00.S.154-3.S-ext, status: n/a
File: bootflash:Active_Dir/asr1000rp2-espbase.03.13.00.S.154-3.S-ext.pkg, on: ESP0
Built: 2013-07-25_21.16, by: mcpre
File SHA1 checksum: 2fe0ede1545e3f8260b7d453653e812500f0d7b0

Package: espx86base, version: 03.13.00.S.154-3.S-ext, status: n/a
File: bootflash:Active_Dir/asr1000rp2-espx86base.03.13.00.S.154-3.S-ext.pkg, on: ESP0
Built: 2013-07-25_22.55, by: mcpre
File SHA1 checksum: 571b8bb3866341badd6e24de677b98409f0c789c

Package:espbase, version: 03.13.00.S.154-3.S-ext, status: n/a
File: bootflash:Active_Dir/asr1000rp2-espbase.03.13.00.S.154-3.S-ext.pkg, on: ESP1
Built: 2013-07-25_21.16, by: mcpre
File SHA1 checksum: 2fe0ede1545e3f8260b7d453653e812500f0d7b0

Package: espx86base, version: 03.13.00.S.154-3.S-ext, status: n/a
File:bootflash:Active_Dir/asr1000rp2-espx86base.03.13.00.S.154-3.S-ext.pkg, on: ESP1
Built: 2013-07-25_22.55, by: mcpre
File SHA1 checksum: 571b8bb3866341badd6e24de677b98409f0c789c

Package: sipbase,
version: 03.13.00.S.154-3.S-ext, status: n/a
File: bootflash:Active_Dir/asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg, on: SIP0
Built: 2013-07-25_21.16, by: mcpre
File SHA1 checksum: 3b6a4838972840a995ff22e73fd2bae910b268a7

Package: elcbase, version: 03.13.00.S.154-3.S-ext, status: n/a
File: bootflash:Active_Dir/asr1000rp2-elcbase.03.13.00.S.154-3.S-ext.pkg, on: SIP0
Built: 2013-07-25_21.16, by: mcpre
File SHA1 checksum: 99f8dc925083b118626a4e82d93079050db96826

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP0/0
Built: 2013-07-25_21.16, by: mcpre
File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP0/0
Built: 2013-07-25_21.16, by: mcpre
File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP0/1
Built: 2013-07-25_21.16, by: mcpre
File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP0/1
Built: 2013-07-25_21.16, by: mcpre
File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499
```

```
Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP0/2
Built: 2013-07-25_21.16, by: mcpre
File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP0/2
Built: 2013-07-25_21.16, by: mcpre
File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP0/3
Built: 2013-07-25_21.16, by: mcpre
File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP0/3
Built: 2013-07-25_21.16, by: mcpre
File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipbase, version: 03.13.00.S.154-3.S-ext, status: n/a
File: bootflash:Active_Dir/asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg, on: SIP1
Built: 2013-07-25_21.16, by: mcpre
File SHA1 checksum: 3b6a4838972840a995ff22e73fd2bae910b268a7

Package: elcbase, version: 03.13.00.S.154-3.S-ext, status: n/a
File: bootflash:Active_Dir/asr1000rp2-elcbase.03.13.00.S.154-3.S-ext.pkg, on: SIP1
Built: 2013-07-25_21.16, by: mcpre
File SHA1 checksum: 99f8dc925083b118626a4e82d93079050db96826

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP1/0
Built: 2013-07-25_21.16, by: mcpre
File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP1/0
Built: 2013-07-25_21.16, by: mcpre
File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP1/1
Built: 2013-07-25_21.16, by: mcpre
File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP1/1
Built: 2013-07-25_21.16, by: mcpre
File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP1/2
Built: 2013-07-25_21.16, by: mcpre
File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP1/2
Built: 2013-07-25_21.16, by: mcpre
File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP1/3
Built: 2013-07-25_21.16, by: mcpre
```

```
File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP1/3
Built: 2013-07-25_21.16, by: mcpre
File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipbase, version: 03.13.00.S.154-3.S-ext, status: n/a
File: bootflash:Active_Dir/asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg, on: SIP2
Built: 2013-07-25_21.16, by: mcpre
File SHA1 checksum: 3b6a4838972840a995ff22e73fd2bae910b268a7

Package: elcbase, version: 03.13.00.S.154-3.S-ext, status: n/a
File: bootflash:Active_Dir/asr1000rp2-elcbase.03.13.00.S.154-3.S-ext.pkg, on: SIP2
Built: 2013-07-25_21.16, by: mcpre
File SHA1 checksum: 99f8dc925083b118626a4e82d93079050db96826

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP2/0
Built: 2013-07-25_21.16, by: mcpre
File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP2/0
Built: 2013-07-25_21.16, by: mcpre
File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP2/1
Built: 2013-07-25_21.16, by: mcpre
File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP2/1
Built: 2013-07-25_21.16, by: mcpre
File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP2/2
Built: 2013-07-25_21.16, by: mcpre
File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP2/2
Built: 2013-07-25_21.16, by: mcpre
File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP2/3
Built: 2013-07-25_21.16, by: mcpre
File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP2/3
Built: 2013-07-25_21.16, by: mcpre
File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipbase, version: 03.13.00.S.154-3.S-ext, status: n/a
File: bootflash:Active_Dir/asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg, on: SIP3
Built: 2013-07-25_21.16, by: mcpre
File SHA1 checksum: 3b6a4838972840a995ff22e73fd2bae910b268a7

Package: elcbase, version: 03.13.00.S.154-3.S-ext, status: n/a
File: bootflash:Active_Dir/asr1000rp2-lcbase.03.13.00.S.154-3.S-ext.pkg, on: SIP3
```

```
Built: 2013-07-25_21.16, by: mcpre
File SHA1 checksum: 99f8dc925083b118626a4e82d93079050db96826

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP3/0
Built: 2013-07-25_21.16, by: mcpre
File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP3/0
Built: 2013-07-25_21.16, by: mcpre
File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP3/1
Built: 2013-07-25_21.16, by: mcpre
File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP3/1
Built: 2013-07-25_21.16, by: mcpre
File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP3/2
Built: 2013-07-25_21.16, by: mcpre
File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP3/2
Built: 2013-07-25_21.16, by: mcpre
File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP3/3
Built: 2013-07-25_21.16, by: mcpre
File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP3/3
Built: 2013-07-25_21.16, by: mcpre
File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipbase, version: 03.13.00.S.154-3.S-ext, status: n/a
File: bootflash:Active_Dir/asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg, on: SIP4
Built: 2013-07-25_21.16, by: mcpre
File SHA1 checksum: 3b6a4838972840a995ff22e73fd2bae910b268a7

Package: elcbase, version: 03.13.00.S.154-3.S-ext, status: n/a
File: bootflash:Active_Dir/asr1000rp2-elcbase.03.13.00.S.154-3.S-ext.pkg, on: SIP4
Built: 2013-07-25_21.16, by: mcpre
File SHA1 checksum: 99f8dc925083b118626a4e82d93079050db96826

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP4/0
Built: 2013-07-25_21.16, by: mcpre
File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca
Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP4/0
Built: 2013-07-25_21.16, by: mcpre
File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP4/1
```

```
Built: 2013-07-25_21.16, by: mcpre
File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP4/1
Built: 2013-07-25_21.16, by: mcpre
File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP4/2
Built: 2013-07-25_21.16, by: mcpre
File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP4/2
Built: 2013-07-25_21.16, by: mcpre
File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP4/3
Built: 2013-07-25_21.16, by: mcpre
File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP4/3
Built: 2013-07-25_21.16, by: mcpre
File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipbase, version: 03.13.00.S.154-3.S-ext, status: n/a
File: bootflash:Active_Dir/asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg, on: SIP5
Built: 2013-07-25_21.16, by: mcpre
File SHA1 checksum: 3b6a4838972840a995ff22e73fd2bae910b268a7

Package: elcbase, version: 03.13.00.S.154-3.S-ext, status: n/a
File: bootflash:Active_Dir/asr1000rp2-elcbase.03.13.00.S.154-3.S-ext.pkg, on: SIP5
Built: 2013-07-25_21.16, by: mcpre
File SHA1 checksum: 99f8dc925083b118626a4e82d93079050db96826

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP5/0
Built: 2013-07-25_21.16, by: mcpre
File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP5/0
Built: 2013-07-25_21.16, by: mcpre
File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP5/1
Built: 2013-07-25_21.16, by: mcpre
File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP5/1
Built: 2013-07-25_21.16, by: mcpre
File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499
Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP5/2
Built: 2013-07-25_21.16, by: mcpre
File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP5/2
```

```
Built: 2013-07-25_21.16, by: mcpre
File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP5/3
Built: 2013-07-25_21.16, by: mcpre
File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP5/3
Built: 2013-07-25_21.16, by: mcpre
File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Router# reload
<some output removed for brevity>


Router# request platform software package clean
Cleaning up unnecessary package files

No path specified, will use booted path bootflash:Active_Dir/packages.conf
Cleaning bootflash:Active_Dir
  Scanning boot directory for packages ... done.
  Preparing packages list to delete ...
  asr1000rp2-elcbase.03.13.00.S.154-3.S-ext.pkg

File is in use, will not delete.
    asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg

File is in use, will not delete.
    asr1000rp2-espbase.03.13.00.S.154-3.S-ext.pkg

File is in use, will not delete.
    asr1000rp2-espx86base.03.13.00.S.154-3.S-ext.pkg

File is in use, will not delete.
    asr1000rp2-rpaccess.03.13.00.S.154-3.S-ext.pkg

File is in use, will not delete.
    asr1000rp2-rpbase.03.13.00.S.154-3.S-ext.pkg

File is in use, will not delete.
    asr1000rp2-rpcontrol.03.13.00.S.154-3.S-ext.pkg

File is in use, will not delete.
    asr1000rp2-rpios-adventerprisek9.03.13.00.S.154-3.S-ext.pkg

File is in use, will not delete.
    asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg

 File is in use, will not delete.
    asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg

File is in use, will not delete.
    packages.conf

File is in use, will not delete.
  done.

Files that will be deleted:
  asr1000rp2-elcbase.03.12.01.S.154-2.S.pkg
  asr1000rp2-elcspa.03.12.01.S.154-2.S.pkg
  asr1000rp2-espbase.03.12.01.S.154-2.S.pkg
  asr1000rp2-espx86base.03.12.01.S.154-2.S.pkg
```

```
  asr1000rp2-packages-adventerprisek9.03.12.01.S.154-2.S.conf
  asr1000rp2-rpaccess.03.12.01.S.154-2.S.pkg
  asr1000rp2-rpbase.03.12.01.S.154-2.S.pkg
  asr1000rp2-rpcontrol.03.12.01.S.154-2.S.pkg
  asr1000rp2-rpios-adventerprisek9.03.12.01.S.154-2.S.pkg
  asr1000rp2-sipbase.03.12.01.S.154-2.S.pkg
  asr1000rp2-sipspa.03.12.01.S.154-2.S.pkg
  packages.conf.00-
  packages.conf.01-
  packages.conf.02-
  packages.conf.03-
  packages.conf.04-
  packages.conf.05-

Do you want to proceed? [confirm]y
  Deleting file bootflash:Active_Dir/asr1000rp2-elcbase.03.12.01.S.154-2.S.pkg ... done.
  Deleting file bootflash:Active_Dir/asr1000rp2-elcspa.03.12.01.S.154-2.S.pkg ... done.
  Deleting file bootflash:Active_Dir/asr1000rp2-espbase.03.12.01.S.154-2.S.pkg ... done.
  Deleting file bootflash:Active_Dir/asr1000rp2-espx86base.03.12.01.S.154-2.S.pkg ... doe.
  Deleting file
bootflash:Active_Dir/asr1000rp2-packages-adventerprisek9.03.12.01.S.154-2.S.conf ... done.
  Deleting file bootflash:Active_Dir/asr1000rp2-rpaccess.03.12.01.S.154-2.S.pkg ... done.
  Deleting file bootflash:Active_Dir/asr1000rp2-rpbase.03.12.01.S.154-2.S.pkg ... done.
  Deleting file bootflash:Active_Dir/asr1000rp2-rpcontrol.03.12.01.S.154-2.S.pkg ... done.
  Deleting file
bootflash:Active_Dir/asr1000rp2-rpios-adventerprisek9.03.12.01.S.154-2.S.pkg ... done.
  Deleting file bootflash:Active_Dir/asr1000rp2-sipbase.03.12.01.S.154-2.S.pkg ... done.
  Deleting file bootflash:Active_Dir/asr1000rp2-sipspa.03.12.01.S.154-2.S.pkg ... done.
  Deleting file bootflash:Active_Dir/packages.conf.00- ... done.
  Deleting file bootflash:Active_Dir/packages.conf.01- ... done.
  Deleting file bootflash:Active_Dir/packages.conf.02- ... done.
  Deleting file bootflash:Active_Dir/packages.conf.03- ... done.
  Deleting file bootflash:Active_Dir/packages.conf.04- ... done.
  Deleting file bootflash:Active_Dir/packages.conf.05- ... done.
SUCCESS: Files deleted.
```

Using Subpackages for Software Upgrade on a Cisco ASR 1002 Router or Cisco ASR 1004 Router (request platform Command Set)

This section provides instructions on using software upgrade for Cisco ASR 1002 or 1004 Router running subpackages using the **request platform** command set.

These instructions assume two IOS processes are active on the RP and that the router is already running using subpackages. For information on checking and configuring two IOS processes on the same RP, see the "Using Subpackages for Software Upgrade on a Cisco ASR 1002 Router or Cisco ASR 1004 Router (software upgrade Command Set)" section on page 5-91.

SUMMARY STEPS

1. **show version**

   show version installed

   dir *filesystem*:<*directory*>

   show platform

   show redundancy-states
2. **redundancy**

   mode sso

3. **mkdir** *URL-to-directory-name*
4. **ip tftp** *source-interface gigabitethernet port*
5. **copy tftp:** *URL-to-target-location*
6. **request platform software package expand file** *URL-to-consolidated-package*
7. **dir** *URL-to-consolidated-package*
8. **copy** *file-system*:**asr1000rp2-espbase.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*

   copy *file-system*:**asr1000rp2-espx86base.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*

   copy *file-system*:**asr1000rp2-rpaccess.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*

   copy *file-system*:**asr1000rp2-rpbase.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*

   copy *file-system*:**asr1000rp2-rpcontrol.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*

   copy *file-system*:**asr1000rp2-rpios.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*

   copy *file-system*:**asr1000rp2-sipbase.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*

   copy *file-system*:**asr1000rp2-sipspa.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*

   Copy *file-system*:**asr1000rp2-elcbase.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*

   copy *file-system*:**asr1000rp2-elcspa.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*
9. **request platform software package install rp 0 file** *file-system*:**asr1000rp2-{rpaccess,rpios,rpcontrol}***version-string***.pkg bay** *standby-bay* **force**
10. **redundancy force-switchover**
11. **show platform**
12. **request platform software package install rp 0 file** *file-system*:**asr1000rp2-{rpaccess,rpios,rpcontrol}***version-string***.pkg bay** *standby-bay* **force**
13. **request platform software package install rp 0 file** *file-system*:**asr1000rp2-{sipbase,sipspa}***version***.pkg slot** *SIP-slot-number*

    Repeat the step 13, for each available SIP installed in the router before moving onto the next step.
14. **request platform software package install rp 0 file** *file-system*:**asr1000rp2-{elcbase,elcspa}***version***.pkg slot** *ELC-slot-number*

    Repeat the step 14, for each available ELC installed in the router before moving onto the next step.
15. **request platform software package install rp 0 file** *file-system*:**asr1000rp2-esp***version***.pkg force**
16. **request platform software package install rp 0 file** *file-system*:**asr1000rp***version***.pkg**
17. **show version provisioned**
18. **reload**
19. **request platform software package clean**

Note Step 17, 18, and 19 does not have to be performed immediately, and can be done at a convenient time.

DETAILED STEPS

	Command or Action	Purpose

Step 1	**show version**
	show version installed
	dir *filesystem:<directory>*
	show platform
	show redundancy-states
	
	Example:
	`Router# show version`
	`Router# show version r0 installed`
	`Router# dir bootflash:`
	`Router# show platform`
	`Router# show redundancy-states`
		(Optional) Use the following commands to confirm the current router configuration, as follows:
		• **show version** and **show version** *active-rp* **installed**—Verify the running version of the Cisco IOS XE software on the router, and which file was used to boot the router, and where that file is stored.
		• **dir**—Confirm that the files that were used to boot the router are located in the directory.
		• **show platform**—Confirm the current status of the active and standby RPs.
		• **show redundancy states**—Confirm the operational and configured redundancy states.

Step 2	**redundancy**
	mode sso
	
	Example:
	`Router(config)# redundancy`
	`Router(config-red)# mode sso`
		Configure SSO if it is not already configured.
		Note Save the configuration after making this configuration step.

Step 3	**mkdir** *URL-to-directory-name* **Example:** `Router# mkdir usb0:221subs`	Create a directory to store the consolidated package and subpackages. This directory must be created in most cases because the consolidated packages and subpackages have to be separated from the subpackages that booted the router at this step of the procedure.
Step 4	**ip tftp** *source-interface gigabitethernet port* **Example:** `Router(config)# ip tftp source-interface gigabitethernet 0`	Specifies the Gigabit Ethernet TFTP source-interface to be configured: *slot/port*—Specifies the location of the TFTP source-interface. **Note** To copy a file using TFTP through the Management Ethernet interface, the **ip tftp source-interface GigabitEthernet 0** command must be entered before entering the **copy tftp** command.
Step 5	**copy tftp:** *URL-to-target-location* **Example:** `Router# copy tftp: usb0:221subs`	Copy the consolidated package file into the directory created in Step 3. The consolidated package in this step should not be copied into the same directory where the subpackages that are currently running your router are stored (the directory containing the packages.conf provisioning file from which the router was booted). **Tip** It is recommended that you copy the package onto a usb: or harddisk: file system for space considerations when performing this step of the procedure.

	Command or Action	Purpose
Step 6	**request platform software package expand file** *URL-to-consolidated-package* **Example:** `Router# request platform software package expand file usb0:221subs/asr1000rp2-adventerprisek9.03.13.00.S.154-3.S-ext.bin`	Extract the subpackages out of the consolidated package file into the temporary directory. **Note** Take extra care to extract the subpackages to a temporary subdirectory and do not delete any of the files currently running the router at this point of the procedure. To erase the files that were running on the router before the ISSU upgrade, enter the **request platform software package clean** command after the ISSU upgrade has been completed.
Step 7	**dir** *target-URL* **Example:** `Router# dir usb0:221subs`	(Optional) Display the directory to confirm that the files were extracted.

Command or Action	Purpose

Step 8 **copy** *file-system*:**asr1000rp2-espbase.**_version_.**pkg** *URL-to-directory-of-sub-packages-active-RP*
copy *file-system*:**asr1000rp2-espx86base.**_version_.**pkg** *URL-to-directory-of-sub-packages-active-RP*
copy *file-system*:**asr1000rp2-rpaccess.**_version_.**pkg** *URL-to-directory-of-sub-packages-active-RP*
copy *file-system*:**asr1000rp2-rpbase.**_version_.**pkg** *URL-to-directory-of-sub-packages-active-RP*
copy *file-system*:**asr1000rp2-rpcontrol.**_version_.**pkg** *URL-to-directory-of-sub-packages-active-RP*
copy *file-system*:**asr1000rp2-rpios.**_version_.**pkg** *URL-to-directory-of-sub-packages-active-RP*
copy *file-system*:**asr1000rp2-sipbase.**_version_.**pkg** *URL-to-directory-of-sub-packages-active-RP*
copy *file-system*:**asr1000rp2-sipspa.**_version_.**pkg** *URL-to-directory-of-sub-packages-active-RP*
copy *file-system*:**asr1000rp2-elcbase.**_version_.**pkg** *URL-to-directory-of-sub-packages-active-RP*
copy *file-system*:**asr1000rp2-elcspa.**_version_.**pkg** *URL-to-directory-of-sub-packages-active-RP*
 	
Example:
`Router# copy usb0:221subs/asr1000rp2-espbase.03.13.00.S.154-3.S-ext.pkg bootflash:`
 	
`Router# copy usb0:221subs/asr1000rp2-espx86base.03.13.00.S.154-3.S-ext.pkg bootflash:`
 	
`Router# copy usb0:221subs/asr1000rp2-rpaccess.03.13.00.S.154-3.S-ext.pkg bootflash:`
 	
`Router# copy usb0:221subs/asr1000rp2-rpbase.03.13.00.S.154-3.S-ext.pkg bootflash:`
 	
`Router# copy usb0:221subs/asr1000rp2-rpcontrol.03.13.00.S.154-3.S-ext.pkg bootflash:`
 	
`Router# copy usb0:221subs/asr1000rp2-rpios-adventerprisek.9.03.13.00.S.154-3.S-ext.pkg bootflash:`
 	
`Router# copy usb0:221subs/asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg bootflash:`
 	
`Router# copy usb0:221subs/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg bootflash:`
 	
`Router# copy usb0:221subs/asr1000rp2-elcbase.03.13.00.S.154-3.S-ext.pkg bootflash:`
 	
`Router# copy usb0:221subs/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg bootflash:`
	Copy the subpackages out of the temporary directory into the directory on the router where the subpackages running the active RP are currently stored.
	
	Note Make sure to copy all the subpackage files extracted in step 6 to the current active packages directory.

	Command or Action	Purpose
Step 9	**request platform software package install rp 0 file** *file-system*:**asr1000rp2-{rpaccess,rpios,rpcontrol}*** *version-string*.**pkg bay** *standby-bay* **force** **Example:** `Router# request platform software package install rp 0 file bootflash:asr1000rp2-{rpaccess,rpios,rpcontrol}*03.13.00.S.154-3.S-ext*.pkg bay 1 force`	Upgrade the RPAccess, RPIOS, and RPControl subpackages in the standby bay.
Step 10	**redundancy force-switchover** **Example:** `Router# redundancy force-switchover`	Force a switchover from the active IOS process to the standby IOS process. **Note** Your connection to the router often drops and is expected behavior at this point of the procedure in many scenarios. If this step drops your connection to the router, wait a few minutes before reconnecting to the router and then continue to Step 11.
Step 11	(Optional) **show platform** **Example:** `Router# show platform`	(Optional) Monitor system state to ensure both IOS processes are active.
Step 12	**request platform software package install rp 0 file** *file-system*:**asr1000rp2-{rpaccess,rpios,rpcontrol}*** *version-string*.**pkg bay** *standby-bay* **force** **Example:** `Router# request platform software package install rp 0 file bootflash:asr1000rp2-{rpaccess,rpios,rpcontrol}*03.13.00.S.154-3.S-ext*.pkg bay 0 force`	Upgrade the RPAccess, RPIOS, and RPControl subpackages in the standby bay, which in this context is the opposite bay used in Step 9. **Note** The *pattern* options used in this CLI (**rpaccess**, **rpios**, and **rpcontrol**) were introduced in Cisco IOS XE Release 2.1.2 and are not available in previous Cisco IOS XE Releases. See the "ISSU Procedures (Prior to Cisco IOS XE Release 2.1.2)" section on page 5-72 for pre-Cisco IOS XE Release 2.1.2 ISSU upgrade procedures.

	Command or Action	Purpose

Step 13	**request platform software package install rp 0 file** *file-system*:**asr1000rp2-{sipbase,sipspa}***version***.pkg slot** *SIP-slot-number* **force**
	
	Example:
	`Router# request platform software package install rp 0 file bootflash:asr1000rp2-{sipspa,sipbase}*03.13.00.S.154-3.S-ext*.pkg slot 0 force`
	`Router# request platform software package install rp 0 file bootflash:asr1000rp2-{sipspa,sipbase}*03.13.00.S.154-3.S-ext*.pkg slot 1 force`
	`Router# request platform software package install rp 0 file bootflash:asr1000rp2-{sipspa,sipbase}*03.13.00.S.154-3.S-ext*.pkg slot 2 force`
		Upgrade the SIP and SPA subpackages for each SIP on the router.
		
		Note This step must be completed one SIP at a time, and repeated for each SIP installed on the router before performing the next step.
		
		Tip You can use the **show ip interface brief** command to identify which slots contain SIPs and SPAs. The interfaces with three numbers (in the form *SIP-number*/*SPA-number*/*interface-number*) identify the SIP and SPA locations in the router.
		
		Note The *pattern* options used in this CLI (**sipbase** and **sipspa**) were introduced in Cisco IOS XE Release 2.1.2 and are not available in previous Cisco IOS XE Releases.
		See the "ISSU Procedures (Prior to Cisco IOS XE Release 2.1.2)" section on page 5-72 for pre-Cisco IOS XE Release 2.1.2 ISSU upgrade procedures.

Step 14	**request platform software package install rp 0 file** *file-system*:**asr1000rp2-{elcase,elcspa}***version***.pkg slot** *SIP-slot-number* **force**
	Repeat this step for each ELC installed in your router before proceeding to the next step.
	
	Example:
	`Router# request platform software package install rp 0 file bootflash:asr1000rp2-{elcspa,elcbase}*03.13.00.S.154-3.S-ext*.pkg slot 0 force`
	`Router# request platform software package install rp 0 file bootflash:asr1000rp2-{elcspa,elcbase}*03.13.00.S.154-3.S-ext*.pkg slot 1 force`
	`Router# request platform software package install rp 0 file bootflash:asr1000rp2-{elcspa,elcbase}*03.13.00.S.154-3.S-ext*.pkg slot 2 force`
		Upgrade the ELC and SPA subpackages for each ELC on the router.
		
		Note This step must be completed for one ELC at a time, and repeated for each ELCinstalled on the router before performing the next step.
		
		Tip You can use the **show ip interface brief** command to identify which slots contain SIPs and SPAs. The interfaces with three numbers (in the form *ELC-number*/*SPA-number*/*interface-number*) identify the ELC and SPA locations in the router.
		
		Note The *pattern* options used in this CLI (**elcbase** and **elcspa**) were introduced in Cisco IOS XE Release 3.10S and are not available in previous Cisco IOS XE Releases.

Step 15	**request platform software package install rp 0 file** *file-system*:**asr1000rp2-esp***version***.pkg force** **Example:** `Router# request platform software package install rp 0 file bootflash:asr1000rp2-esp*03.13.00.S.154-3.S-ext*.pkg force`	Upgrade the ESP Base subpackage.

	Command or Action	Purpose
Step 16	**request platform software package install rp 0 file** *file-system*:**asr1000rp***version***.pkg** **Example:** `Router# request platform software package install rp 0 file bootflash:asr1000rp*03.13.00.S.154-3.S-ext*.pkg`	Upgrade all subpackages, including the RPBase subpackage, which is the last subpackage that needs to be upgraded. **Note** This step is required to ensure that all subpackages on the router were upgraded as part of this procedure, and might upgrade some subpackages that would otherwise be missed in the process.
Step 17	**show version installed** **Example:** `Router# show version installed`	(Optional) Verify that the subpackages are properly installed.
Step 18	**reload** **Example:** `Router# reload`	Reload the RP. **Tip** The router will continue normal operation even without a reload, so you can reload the router during scheduled maintenance or a slower traffic period.
Step 19	**request platform software package clean**	(Optional) Removes the unused subpackages from the router.

Examples

The following example shows the software upgrade for Cisco ASR 1002 or 1004 Router running sub-packages:

```
Router# show version
Cisco IOS Software, IOS-XE Software (X86_64_LINUX_IOSD-ADVENTERPRISEK9-M), Version
15.3(2)S, RELEASE SOFTWARE (fc1)

<output removed for brevity>

System image file is "bootflash:Active_Dir/packages.conf"

<output removed for brevity>
cisco ASR1004 (RP2) processor with 1546489K/6147K bytes of memory.
Processor board ID FOX1339G0QJ
32 Gigabit Ethernet interfaces
2 Ten Gigabit Ethernet interfaces
32768K bytes of non-volatile configuration memory.
388608K bytes of physical memory.
933311K bytes of eUSB flash at bootflash:.
78085207K bytes of SATA hard disk at harddisk:

Configuration register is 0x2102

Router# show version installed
Package: Provisioning File, version: n/a, status: active
  File: bootflash:Active_Dir/packages.conf, on: RP0
  Built: n/a, by: n/a
  File SHA1 checksum: a624f70f68c60292f4482433f43afd92487a55c4
  Package: rpbase, version: 03.12.01.S.154-2.S, status: active

File: bootflash:Active_Dir/asr1000rp2-rpbase.03.12.01.S.154-2.S.pkg, on: RP0
  Built: 2013-03-25_18.48, by: mcpre
  File SHA1 checksum: 3a9675142898cfac350d4e42f0e37bd9f4e48538
  Package: rpcontrol, version: 03.12.01.S.154-2.S, status: active
```

```
File: bootflash:Active_Dir/asr1000rp2-rpcontrol.03.12.01.S.154-2.S.pkg, on: RP0/0
  Built: 2013-03-25_18.48, by: mcpre
  File SHA1 checksum: 87b11f863f67fdf2610ee0769b929baab4c3efad


<output removed for brevity>


Router# show redundancy states
    my state = 13 -ACTIVE
    peer state = 8  -STANDBY HOT
    Mode = Duplex
    Unit = Primary
    Unit ID = 48
    Redundancy Mode (Operational) = sso
    Redundancy Mode (Configured)  = sso
    Redundancy State              = sso
    Maintenance Mode = Disabled
    Manual Swact = enabled
    Communications = Up
    client count = 107
    client_notification_TMR = 30000 milliseconds
    RF debug mask = 0x0

Router# show platform
Chassis type: ASR1004

Slot        Type                State                 Insert time (ago)
--------- ------------------- --------------------- -----------------
 0         ASR1000-2T+20X1GE      ok                    00:04:19
 0/0       BUILT-IN-2T+20X1GE     ok                    00:02:36
 1         ASR1000-SIP10          ok                    00:04:19
 1/0       SPA-2X1GE-V2           ok                    00:03:07
 1/1       SPA-10X1GE-V2          ok                    00:03:00
 R0        ASR1000-RP2            ok                    00:04:19
 R0/0                             ok, active            00:04:19
 R0/1                             ok, standby           00:02:41
 F0        ASR1000-ESP40          ok, active            00:04:19
 P0        ASR1004-PWR-AC         ok                    00:03:28
 P1        ASR1004-PWR-AC         ps, fail              00:03:28


Slot      CPLD Version        Firmware Version
--------- ------------------- ----------------------------------------
 0         00200800            15.2(1r)S
 1         07091401            15.3(3r)S
 R0        08103002            15.3(3r)S
 F0        1003190E            15.3(3r)S



Router# mkdir bootflash:harddisk:TargetS_Subs
Create directory filename [Target_Subs]?
Created dir harddisk:/Target_Subs

Router# conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# ip tftp source-interface Gigabitethernet 0

Router(config)# end

Router#copy tftp: harddisk:Target_Subs
```

```
Address or name of remote host []? 202.153.144.25
Source filename []?
/auto/tftp-srg-india/asr1000rp2-adventerprisek9.03.13.00.S.154-3.S-ext.bin
Destination filename [/Target_Subs/asr1000rp2-adventerprisek9.03.13.00.S.154-3.S-ext.bin]?
Accessing
tftp://202.153.144.25//auto/tftp-srg-india/asr1000rp2-adventerprisek9.03.13.00.S.154-3.S-e
xt.bin...
Loading /auto/tftp-srg-india/asr1000rp2-adventerprisek9.03.13.00.S.154-3.S-ext.bin from
202.153.144.25 (via GigabitEthernet0):
!!!!!!!!!!!!!!!!O!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!O!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!O!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
[OK - 569597380 bytes]

569597380 bytes copied in 101.618 secs (5605280 bytes/sec)


Router# request platform software package expand file
harddisk:/Target_Subs/asr1000rp2-adventerprisek9.03.13.00.S.154-3.S-ext.bin

Verifying parameters
Validating package type
Copying package files
SUCCESS: Finished expanding all-in-one software package.

Router# dir harddisk:Target_Subs
Directory of harddisk:/Target_Subs
9666562  -rw-   569597380   Aug 3 2013 13:25:06 +05:30
asr1000rp2-adventerprisek9.03.13.00.S.154-3.S-ext.bin
9666566  -rw-    37557200   Aug 3 2013 13:27:06 +05:30
asr1000rp2-elcbase.03.13.00.S.154-3.S-ext.pkg
9666567  -rw-    51194832   Aug 3 2013 13:27:07 +05:30
asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg
9666568  -rw-    80657364   Aug 3 2013 13:27:07 +05:30
asr1000rp2-espbase.03.13.00.S.154-3.S-ext.pkg
9666569  -rw-    95446456   Aug 3 2013 13:27:08 +05:30
asr1000rp2-espx86base.03.13.00.S.154-3.S-ext.pkg
9666564  -rw-        9381   Aug 3 2013 13:27:06 +05:30
asr1000rp2-packages-adventerprisek9.03.13.00.S.154-3.S-ext.conf
9666570  -rw-    23350232   Aug 3 2013 13:27:08 +05:30
asr1000rp2-rpaccess.03.13.00.S.154-3.S-ext.pkg
9666571  -rw-    37694900   Aug 3 2013 13:27:08 +05:30
asr1000rp2-rpbase.03.13.00.S.154-3.S-ext.pkg
9666572  -rw-    45536216   Aug 3 2013 13:27:08 +05:30
asr1000rp2-rpcontrol.03.13.00.S.154-3.S-ext.pkg
9666573  -rw-   118754284   Aug 3 2013 13:27:08 +05:30
asr1000rp2-rpios-adventerprisek9.03.13.00.S.154-3.S-ext.pkg
9666574  -rw-    38380500   Aug 3 2013 13:27:08 +05:30
asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg
9666575  -rw-    61760468   Aug 3 2013 13:27:08 +05:30
asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg
9666565  -rw-       10165   Aug 3 2013 13:27:08 +05:30  packages.conf
78704144384 bytes total (47541493760 bytes free)


Router# copy harddisk:Target_Subs/asr1000rp2-espbase.03.13.00.S.154-3.S-ext.pkg bootflash:
Active_Dir/Destination filename
[Active_Dir/asr1000rp2-espbase.03.13.00.S.154-3.S-ext.pkg]?
Copy in
progress...CCCC
```

```
Router# copy harddisk:Target_Subs/asr1000rp2-espx86base.03.13.00.S.154-3.S-ext.pkg
bootflash:
Active_Dir/Destination filename
[Active_Dir/asr1000rp2-espx86base.03.13.00.S.154-3.S-ext.pkg]?
Copy in
progress...CCCC
```

```
Router# copy harddisk:Target_Subs/asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg bootflash:
Active_Dir/Destination filename
[Active_Dir/asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg]?
Copy in
progress...CCCC
```

```
Processing image file constraints
Creating candidate provisioning file
Finished image file verification
--- Starting candidate package set construction ---
Verifying existing software set
Processing candidate provisioning file
Constructing working set for candidate package set
Constructing working set for running package set
Checking command output
Constructing merge of running and candidate packages
Checking if resulting candidate package set would be complete
Finished candidate package set construction
--- Starting compatibility testing ---
Determining whether candidate package set is compatible
WARNING:
WARNING: Candidate software combination not found in compatibility database
WARNING:

Determining whether installation is valid
Creating matrix_file by locate_latest_matrix_file  /tmp/issu/provision/s
WARNING:
WARNING: Candidate software combination not found in compatibility database
WARNING:
WARNING:
WARNING: Candidate software combination not found in compatibility database
WARNING:
Software sets are identified as compatible
Verifying image type compatibility
Checking IPC compatibility with running software
Checking candidate package set infrastructure compatibility
Checking infrastructure compatibility with running software
Checking package specific compatibility
Finished compatibility testing
--- Starting impact testing---
Checking operational impact of change

WARNING: Connection may be lost during installation of IOS package
Finished impact testing
--- Starting list of software package changes ---
No old package files removed
New files list:
  Added asr1000rp2-rpaccess.03.13.00.S.154-3.S-ext.pkg
  Added asr1000rp2-rpcontrol.03.13.00.S.154-3.S-ext.pkg
  Added asr1000rp2-rpios-adventerprisek9.03.13.00.S.154-3.S-ext.pkg
Finished list of software package changes
--- Starting commit of software changes ---
Updating provisioning rollback files
Creating pending provisioning file
Committing provisioning file
Finished commit of software changes
--- Starting analysis of software changes ---
Finished analysis of software changes
--- Starting update running software --
Blocking peer synchronization of operating information
Creating the command set placeholder directory
  Finding latest command set
  Finding latest command shortlist lookup file
  Finding latest command shortlist file
  Assembling CLI output libraries
  Assembling CLI input libraries
  Assembling Dynamic configuration files
  Applying interim IPC and database definitions
  Replacing running software
  Replacing CLI software
```

```
  Restarting software
  Restarting IOS PID: 21552, in slot/bay 0/1
  Applying final IPC and database definitions
  Generating software version information
  Notifying running software of updates
  Unblocking peer synchronization of operating information
  Unmounting old packages
  Cleaning temporary installation files
  Finished update running software


SUCCESS: Finished installing software.


Router# redundancy force-switchover
Proceed with switchover to standby RP? [confirm]
   Manual Swact = enabled
%IOSXE_INFRA-6-CONSOLE_ACTIVE: R0/1 console active.
Press RETURN to get started!
[OK]
*Aug  3 13:43:52.101 IST: %CMANRP-6-CMHASTATUS: RP switchover, received chassis event to
become active
*Aug  3 13:43:52.193 IST: %REDUNDANCY-3-SWITCHOVER: RP switchover (PEER_NOT_PRESENT)
*Aug  3 13:43:52.194 IST: %REDUNDANCY-3-SWITCHOVER: RP switchover (PEER_DOWN)
*Aug  3 13:43:52.194 IST: %REDUNDANCY-3-SWITCHOVER: RP switchover
(PEER_REDUNDANCY_STATE_CHANGE)
*Aug  3 13:43:51.231 IST: %RF-5-RF_RELOAD: Self reload. Reason: redundancy
force-switchover
*Aug  3 13:43:51.461 IST: %SYS-5-SWITCHOVER: Switchover requested by
red_switchover_process. Reason: redundancy force-switchover.
*Aug  3 13:43:52.198 IST: %PLATFORM-6-HASTATUS: RP switchover, sent message became active.
IOS is ready to switch to primary after chassis confirmation
*Aug  3 13:43:52.200 IST: %CMANRP-6-CMHASTATUS: RP switchover, received chassis event
became active
*Aug  3 13:43:52.449 IST: %PLATFORM-6-HASTATUS_DETAIL: RP switchover, received chassis
event became active. Switch to primary (count 1)
*Aug  3 13:43:52.733 IST: %LINEPROTO-5-UPDOWN: Line protocol on Interface
GigabitEthernet0, changed state to down
*Aug  3 13:43:53.098 IST: % Redundancy mode change to SSO
*Aug  3 13:43:53.126 IST: %LINK-3-UPDOWN: Interface Lsmpi0, changed state to up
*Aug  3 13:43:53.127 IST: %LINK-3-UPDOWN: Interface EOBC0, changed state to up
*Aug  3 13:43:53.127 IST: %LINK-3-UPDOWN: Interface LIIN0, changed state to up
*Aug  3 13:43:54.127 IST: %LINEPROTO-5-UPDOWN: Line protocol on Interface Lsmpi0, changed
state to up
*Aug  3 13:43:54.127 IST: %LINEPROTO-5-UPDOWN: Line protocol on Interface EOBC0, changed
state to up
*Aug  3 13:43:54.127 IST: %LINEPROTO-5-UPDOWN: Line protocol on Interface LIIN0, changed
state to up
*Aug  3 13:43:55.117 IST: %LINK-3-UPDOWN: Interface Null0, changed state to up
*Aug  3 13:43:55.117 IST: %LINK-3-UPDOWN: Interface GigabitEthernet0/0/8, changed state to
up
*Aug  3 13:43:55.117 IST: %LINK-3-UPDOWN: Interface GigabitEthernet0/0/10, changed state
to up
*Aug  3 13:43:55.117 IST: %LINK-3-UPDOWN: Interface GigabitEthernet1/0/0, changed state to
up
*Aug  3 13:43:55.119 IST: %LINK-3-UPDOWN: Interface GigabitEthernet1/0/1, changed state to
up
*Aug  3 13:43:55.119 IST: %LINK-3-UPDOWN: Interface GigabitEthernet1/1/0, changed state to
up
*Aug  3 13:43:55.119 IST: %LINK-3-UPDOWN: Interface GigabitEthernet1/1/1, changed state to
up
*Aug  3 13:43:55.119 IST: %LINK-3-UPDOWN: Interface GigabitEthernet1/1/3, changed state to
up
```

```
*Aug  3 13:43:55.119 IST: %LINK-3-UPDOWN: Interface GigabitEthernet1/1/5, changed state to
up
*Aug  3 13:43:55.124 IST: %LINK-5-CHANGED: Interface GigabitEthernet0/0/0, changed state
to administratively down
*Aug  3 13:43:55.124 IST: %LINK-5-CHANGED: Interface GigabitEthernet0/0/1, changed state
to administratively down
*Aug  3 13:43:55.124 IST: %LINK-5-CHANGED: Interface GigabitEthernet0/0/2, changed state
to administratively down
*Aug  3 13:43:55.124 IST: %LINK-5-CHANGED: Interface GigabitEthernet0/0/3, changed state
to administratively down
*Aug  3 13:43:55.124 IST: %LINK-5-CHANGED: Interface GigabitEthernet0/0/4, changed state
to administratively down
*Aug  3 13:43:55.124 IST: %LINK-5-CHANGED: Interface GigabitEthernet0/0/5, changed state
to administratively down
*Aug  3 13:43:55.124 IST: %LINK-5-CHANGED: Interface GigabitEthernet0/0/6, changed state
to administratively down
*Aug  3 13:43:55.124 IST: %LINK-5-CHANGED: Interface GigabitEthernet0/0/7, changed state
to administratively down
*Aug  3 13:43:55.124 IST: %LINK-5-CHANGED: Interface GigabitEthernet0/0/9, changed state
to administratively down
*Aug  3 13:43:55.124 IST: %LINK-5-CHANGED: Interface GigabitEthernet0/0/11, changed state
to administratively down
*Aug  3 13:43:56.118 IST: %LINEPROTO-5-UPDOWN: Line protocol on Interface Null0, changed
state to up
*Aug  3 13:43:56.118 IST: %LINEPROTO-5-UPDOWN: Line protocol on Interface
GigabitEthernet0/0/8, changed state to up
*Aug  3 13:43:56.118 IST: %LINEPROTO-5-UPDOWN: Line protocol on Interface
GigabitEthernet0/0/10, changed state to up
*Aug  3 13:43:56.118 IST: %LINEPROTO-5-UPDOWN: Line protocol on Interface
TenGigabitEthernet0/0/20, changed state to down
*Aug  3 13:43:56.118 IST: %LINEPROTO-5-UPDOWN: Line protocol on Interface
TenGigabitEthernet0/0/21, changed state to down
*Aug  3 13:43:56.118 IST: %LINEPROTO-5-UPDOWN: Line protocol on Interface
GigabitEthernet1/0/0, changed state to up
*Aug  3 13:43:56.120 IST: %LINEPROTO-5-UPDOWN: Line protocol on Interface
GigabitEthernet1/0/1, changed state to up
*Aug  3 13:43:56.120 IST: %LINEPROTO-5-UPDOWN: Line protocol on Interface
GigabitEthernet1/1/0, changed state to up
*Aug  3 13:43:56.120 IST: %LINEPROTO-5-UPDOWN: Line protocol on Interface
GigabitEthernet1/1/1, changed state to up
*Aug  3 13:43:56.120 IST: %LINEPROTO-5-UPDOWN: Line protocol on Interface
GigabitEthernet1/1/3, changed state to up
*Aug  3 13:44:05.127 IST: %LINK-3-UPDOWN: Interface GigabitEthernet0, changed state to up
*Aug  3 13:44:06.127 IST: %LINEPROTO-5-UPDOWN: Line protocol on Interface
GigabitEthernet0, changed state to up
*Aug  3 13:44:32.069 IST: %REDUNDANCY-5-PEER_MONITOR_EVENT: Active detected a standby
insertion (raw-event=PEER_FOUND(4))
*Aug  3 13:44:32.070 IST: %REDUNDANCY-5-PEER_MONITOR_EVENT: Active detected a standby
insertion (raw-event=PEER_REDUNDANCY_STATE_CHANGE(5))
*Aug  3 13:44:34.969 IST: %REDUNDANCY-3-IPC: IOS versions do not match.
*Aug  3 13:45:40.946 IST: %RF-5-RF_TERMINAL_STATE: Terminal state reached for (SSO)


Router# show platform
Chassis type: ASR1004

Slot      Type                State                   Insert time (ago)
--------- ------------------- ---------------------- -----------------
 0         ASR1000-2T+20X1GE    ok                      00:29:33
 0/0       BUILT-IN-2T+20X1GE   ok                      00:05:28
 1         ASR1000-SIP10        ok                      00:29:33
 1/0       SPA-2X1GE-V2         ok                      00:05:28
  1/1       SPA-10X1GE-V2        ok                       00:05:28
  R0        ASR1000-RP2          ok                       00:29:33
```

```
  R0/0                           ok, standby           00:01:46
  R0/1                           ok, active            00:06:12
  F0        ASR1000-ESP40        ok, active            00:29:33
  P0        ASR1004-PWR-AC       ok                    00:28:42
  P1        ASR1004-PWR-AC       ps, fail              00:28:42


Slot      CPLD Version        Firmware Version
--------- ------------------- ----------------------------------------
 0         00200800            15.2(1r)S
 1         07091401            15.3(3r)S
 R0        08103002            15.3(3r)S
 F0        1003190E            15.3(3r)S


Router# request platform software package install rp 0 file
bootflash:Active_Dir/asr1000rp2-{rpaccess,rpios,rpcontrol}*03.13.00.S.154-3.S-ext*.pkg bay
0 force
--- Starting local lock acquisition on R0 ---
Finished local lock acquisition on R0
--- Starting file path checking---
Finished file path checking
--- Starting image file verification ---
Checking image file names
Locating image files and validating name syntax
  Found asr1000rp2-rpaccess.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-rpios-adventerprisek9.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-rpcontrol.03.13.00.S.154-3.S-ext.pkg
Verifying image file locations
Inspecting image file types
Processing image file constraints
Creating candidate provisioning file
Finished image file verification
--- Starting candidate package set construction ---
Verifying existing software set
Processing candidate provisioning file
Constructing working set for candidate package set
Constructing working set for running package set
Checking command output
Constructing merge of running and candidate packages
Checking if resulting candidate package set would be complete
Finished candidate package set construction
--- Starting compatibility testing ---
Determining whether candidate package set is compatible
WARNING:
WARNING: Candidate software combination not found in compatibility database
WARNING:Determining whether installation is valid
Creating matrix_file by locate_latest_matrix_file  /tmp/issu/provision/sw
Software sets are identified as compatible
Verifying image type compatibility
Checking IPC compatibility with running software
Checking candidate package set infrastructure compatibility
Checking infrastructure compatibility with running software
Checking package specific compatibility
Finished compatibility testing

--- Starting impact testing ---
Checking operational impact of change
WARNING: Connection may be lost during installation of IOS package
Finished impact testing
--- Starting list of software package changes ---

Old files list:
```

```
Removed asr1000rp2-rpaccess.03.12.01.S.154-2.S.pkg
Removed asr1000rp2-rpcontrol.03.12.01.S.154-2.S.pkg
Removed asr1000rp2-rpios-adventerprisek9.03.12.01.S.154-2.S.pkg
No new package files added
Finished list of software package changes
--- Starting commit of software changes ---
Updating provisioning rollback files
Creating pending provisioning file
Committing provisioning file
Finished commit of software changes
--- Starting analysis of software changes ---
Finished analysis of software changes
--- Starting update running software ---
Blocking peer synchronization of operating information
Creating the command set placeholder directory
  Finding latest command set
  Finding latest command shortlist lookup file
  Finding latest command shortlist file
  Assembling CLI output libraries
  Assembling CLI input libraries
  Assembling Dynamic configuration files
  Applying interim IPC and database definitions
  Replacing running software
  Replacing CLI software
  Restarting software
  Restarting IOS PID: 25452, in slot/bay 0/0
*Aug  3 13:48:07.051 IST: %REDUNDANCY-3-STANDBY_LOST: Standby processor fault
(PEER_NOT_PRESENT)
*Aug  3 13:48:07.052 IST: %REDUNDANCY-3-STANDBY_LOST: Standby processor fault (PEER_DOWN)
*Aug  3 13:48:07.052 IST: %REDUNDANCY-3-STANDBY_LOST: Standby processor fault
(PEER_REDUNDANCY_STATE_CHANGE)
*Aug  3 13:48:09.683 IST: %RF-5-RF_RELOAD: Peer reload. Reason: EHSA standby down
*Aug  3 13:48:09.692 IST: % Redundancy mode change to SSO
**Aug  3 13:48:49.450 IST: %REDUNDANCY-5-PEER_MONITOR_EVENT: Active detected a standby
insertion (raw-event=PEER_FOUND(4))
*Aug  3 13:48:49.451 IST: %REDUNDANCY-5-PEER_MONITOR_EVENT: Active detected a standby
insertion (raw-event=PEER_REDUNDANCY_STATE_CHANGE(5))
*Aug  3 13:48:50.035 IST: %ISSU-3-PEER_IMAGE_NOT_IN_INCOMP_LIST: Peer image
(X86_64_LINUX_IOSD-ADVENTERPRISEK9-M), version (15.3(3)S) on peer uid (48) is not
in the incompatible images list
*Aug  3 13:48:50.036 IST: %ISSU-3-PEER_IMAGE_NOT_IN_INCOMP_LIST: Peer image
(X86_64_LINUX_IOSD-ADVENTERPRISEK9-M), version (15.3(3)S) on peer uid (48) is not
in the incompatible images list  Applying final IPC and database definitions
*Aug  3 13:48:53.521 IST: %DYNCMD-7-CMDSET_UNLOADED: The Dynamic Command set from the
Shell Manager has been unloaded
Generating software version information
Notifying running software of updates
Unblocking peer synchronization of operating information
Unmounting old packages
cleaning temporary installation files
Finished update running software

SUCCESS: Finished installing software.

*Aug  3 13:49:57.477 IST: %NBAR_HA-5-NBAR_INFO: NBAR sync DONE!

*Aug  3 13:49:57.893 IST: %ISSU-3-PEER_IMAGE_REM_FROM_INCOMP_LIST: Peer image
(X86_64_LINUX_IOSD-ADVENTERPRISEK9-M), version (15.3(2)S) on peer uid (48)
being removed from the incompatibility list
*Aug  3 13:49:57.893 IST: %HA_CONFIG_SYNC-6-BULK_CFGSYNC_SUCCEED: Bulk Sync succeeded
*Aug  3 13:49:57.895 IST: %RF-5-RF_TERMINAL_STATE: Terminal state reached for (SSO)

Router # issu commitversion
--- Starting local lock acquisition on R0 ---
```

```
Finished local lock acquisition on R0
--- Starting installation changes ---
Cancelling rollback timer
Finished installation changes

SUCCESS: Installation changes committed


Router# request platform software package install rp 0 file
bootflash:Active_Dir/asr1000rp2-{sipbase,sipspa}*03.13.00.S.154-3.S-ext*.pkg slot 1 force
--- Starting local lock acquisition on R0 ---
Finished local lock acquisition on R0
--- Starting file path checking ---
Finished file path checking
--- Starting image file verification---
Checking image file names
Locating image files and validating name syntax
 Found asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg
 Found asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg
Verifying image file locations
Inspecting image file types
Processing image file constraints
Creating candidate provisioning file
Finished image file verification
-- Starting candidate package set construction ---
Verifying existing software set
Processing candidate provisioning file
Constructing working set for candidate package set
Constructing working set for running package set
Checking command output
Constructing merge of running and candidate packages
Checking if resulting candidate package set would be complete
Finished candidate package set construction
--- Starting compatibility testing ---
Determining whether candidate package set is compatible
WARNING: Candidate software combination not found in compatibility database
WARNING: Determining whether installation is valid
Creating matrix_file by locate_latest_matrix_file  /tmp/issu/provision/s
WARNING: Candidate software combination not found in compatibility database
WARNING: Candidate software combination not found in compatibility database
WARNING: Software sets are identified as compatible
Verifying image type compatibility
Checking IPC compatibility with running software
Checking candidate package set infrastructure compatibility
Checking infrastructure compatibility with running software
Checking package specific compatibility
Finished compatibility testing

--- Starting impact testing ---
Checking operational impact of change
Finished impact testing
--- Starting list of software package changes---
No old package files removed
New files list:
  Added asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg
  Added asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg
Finished list of software package changes
--- Starting commit of software changes ---
Updating provisioning rollback files
Creating pending provisioning file
Committing provisioning file
Finished commit of software changes
--- Starting analysis of software changes ---
Finished analysis of software changes
```

```
--- Starting update running software ---
Blocking peer synchronization of operating information
Creating the command set placeholder directory
  Finding latest command set
  Finding latest command shortlist lookup file
  Finding latest command shortlist file
  Assembling CLI output libraries
  Assembling CLI input libraries
  Assembling Dynamic configuration files
Applying interim IPC and database definitions
Replacing running software
Replacing CLI software
Restarting software
Restarting SIP1
Applying final IPC and database definitions
*Aug  3 13:52:05.767 IST: %IOSXE_OIR-6-OFFLINECARD: Card (cc) offline in slot 1
*Aug  3 13:52:05.770 IST: %IOSXE_OIR-6-REMSPA: SPA removed from subslot 1/0, interfaces
disabled
*Aug  3 13:52:05.770 IST: %IOSXE_OIR-6-REMSPA: SPA removed from subslot 1/1, interfaces
disabled
*Aug  3 13:52:05.778 IST: %SPA_OIR-6-OFFLINECARD: SPA (SPA-2X1GE-V2) offline in subslot
1/0
*Aug  3 13:52:05.786 IST: %SPA_OIR-6-OFFLINECARD: SPA (SPA-10X1GE-V2) offline in subslot
1/1
Generating software version information
Notifying running software of updates
Unblocking peer synchronization of operating information
Unmounting old packages
Cleaning temporary installation files
Finished update running software

SUCCESS: Finished installing software.

*Aug  3 13:53:00.262 IST: %IOSXE_OIR-6-ONLINECARD: Card (cc) online in slot 1
*Aug  3 13:53:00.901 IST: %IOSXE_OIR-6-INSSPA: SPA inserted in subslot 1/0
*Aug  3 13:53:01.194 IST: %IOSXE_OIR-6-INSSPA: SPA inserted in subslot 1/1
*Aug  3 13:53:18.148 IST: %LINK-3-UPDOWN: SIP1/0: Interface EOBC1/1, changed state to up
*Aug  3 13:53:23.978 IST: %SPA_OIR-6-ONLINECARD: SPA (SPA-2X1GE-V2) online in subslot 1/0
*Aug  3 13:53:23.609 IST: %LINK-3-UPDOWN: SIP1/1: Interface EOBC1/1, changed state to up
*Aug  3 13:53:26.288 IST: %LINK-3-UPDOWN: Interface GigabitEthernet1/0/0,changed state to
up
*Aug  3 13:53:26.471 IST: %LINK-3-UPDOWN: Interface GigabitEthernet1/0/1, changed state to
up
*Aug  3 13:53:26.605 IST: %LINK-3-UPDOWN: SIP1/0: Interface GigabitEthernet1/0/0, changed
state to up
*Aug  3 13:53:29.056 IST: %SPA_OIR-6-ONLINECARD: SPA (SPA-10X1GE-V2) online in subslot 1/1
*Aug  3 13:53:31.020 IST: %LINK-3-UPDOWN: Interface GigabitEthernet1/1/1, changed state to
up
*Aug  3 13:53:31.030 IST: %LINK-3-UPDOWN: Interface GigabitEthernet1/1/3, changed state to
up
*Aug  3 13:53:31.042 IST: %LINK-3-UPDOWN: Interface GigabitEthernet1/1/4, changed state to
down
*Aug  3 13:53:31.045 IST: %LINK-3-UPDOWN: Interface GigabitEthernet1/1/5, changed state to
up
*Aug  3 13:53:31.046 IST: %LINK-3-UPDOWN: Interface GigabitEthernet1/1/6, changed state to
down
*Aug  3 13:53:31.048 IST: %LINK-3-UPDOWN: Interface GigabitEthernet1/1/7, changed state to
down
*Aug  3 13:53:31.051 IST: %LINK-3-UPDOWN: Interface GigabitEthernet1/1/8, changed state to
down
*Aug  3 13:53:31.053 IST: %LINK-3-UPDOWN: Interface GigabitEthernet1/1/9, changed state to
down
*Aug  3 13:53:31.312 IST: %LINK-3-UPDOWN: Interface GigabitEthernet1/1/0, changed state to
up
```

```
*Aug  3 13:53:32.833 IST: %LINK-3-UPDOWN: SIP1/1: Interface GigabitEthernet1/1/0, changed
state to up
*Aug  3 13:53:33.007 IST: %LINK-3-UPDOWN: SIP1/1: Interface GigabitEthernet1/1/7, changed
state to down



Router# issu commitversion
-- Starting local lock acquisition on R0 ---
Finished local lock acquisition on R0
--- Starting installation changes ---
Cancelling rollback timer
Finished installation changes


SUCCESS: Installation changes committed


Router#
Router# request platform software package install rp 0 file
bootflash:Active_Dir/asr1000rp2-{elcbase,elcspa}*03.13.00.S.154-3.S-ext*.pkg slot 0 force
---- Starting local lock acquisition on R0 ---
Finished local lock acquisition on R0
-- Starting file path checking ---
Finished file path checking
--- Starting image file verification ---
Checking image file names
Locating image files and validating name syntax
  Found asr1000rp2-elcbase.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg
  Verifying image file locations
  Inspecting image file types
  Processing image file constraints
  Creating candidate provisioning file
  Finished image file verification
---Starting candidate package set construction ---
Verifying existing software set
Processing candidate provisioning file
Constructing working set for candidate package set
Constructing working set for running package set
Checking command output
Constructing merge of running and candidate packages
Checking if resulting candidate package set would be complete
Finished candidate package set construction
--- Starting compatibility testing ---
Determining whether candidate package set is compatible
WARNING: Candidate software combination not found in compatibility database
WARNING: Determining whether installation is valid
Creating matrix_file by locate_latest_matrix_file  /tmp/issu/provision/s
WARNING: Candidate software combination not found in compatibility database
WARNING: Candidate software combination not found in compatibility database
WARNING: Software sets are identified as compatible
Verifying image type compatibility
Checking IPC compatibility with running software
Checking candidate package set infrastructure compatibility
Checking infrastructure compatibility with running software
Checking package specific compatibility
Finished compatibility testing
--- Starting impact testing ---
Checking operational impact of change
Finished impact testing
--- Starting list of software package changes --
No old package files removed
New files list:
```

```
  Added asr1000rp2-elcbase.03.13.00.S.154-3.S-ext.pkg
  Added asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg
  Finished list of software package changes
--- Starting commit of software changes ---
Updating provisioning rollback files
Creating pending provisioning file
Committing provisioning file
Finished commit of software changes
--Starting analysis of software changes ---
Finished analysis of software changes
--- Starting update running software ---
Blocking peer synchronization of operating information
Creating the command set placeholder directory
  Finding latest command set
  Finding latest command shortlist lookup file
  Finding latest command shortlist file
  Assembling CLI output libraries
  Assembling CLI input libraries
  Assembling Dynamic configuration files
  Applying interim IPC and database definitions
  Replacing running software
  Replacing CLI software
  Restarting software
  Applying final IPC and database definitions
  Generating software version information
  Notifying running software of updates
  Unblocking peer synchronization of operating information
  Unmounting old packages
  Cleaning temporary installation files
  Finished update running software


SUCCESS: Finished installing software.


Router# issu commitversion
--- Starting locallock acquisition on R0 ---
Finished local lock acquisition on R0
--- Starting installation changes ---
Cancelling rollback timer
Finished installation changes

SUCCESS: Installation changes committed


Router# request platform software package install rp 0 file
bootflash:Active_Dir/asr1000rp2-esp*03.13.00.S.154-3.S-ext*.pkg force
--- Starting local lock acquisition on R0
---Finished local lock acquisition on R0
--- Starting file path checking ---
Finished file path checking
---Starting image file verification---
Checking image file names
Locating image files and validating name syntax
  Found asr1000rp2-espbase.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-espx86base.03.13.00.S.154-3.S-ext.pkg
  Verifying image file locations
  Inspecting image file types
  Processing image file constraint
  Creating candidate provisioning file
  Finished image file verification

--- Starting candidate package set construction ---
Verifying existing software set
```

```
Processing candidate provisioning file
Constructing working set for candidate package set
Constructing working set for running package set
Checking command output
Constructing merge of running and candidate packages
Checking if resulting candidate package set would be complete
Finished candidate package set construction
--- Starting compatibility testing ---
Determining whether candidate package set is compatible
WARNING: Candidate software combination not found in compatibility database
WARNING: Determining whether installation is valid
Creating matrix_file by locate_latest_matrix_file  /tmp/issu/provision/s
WARNING: Candidate software combination not found in compatibility database
WARNING: Candidate software combination not found in compatibility database
WARNING: Software sets are identified as compatible
Verifying image type compatibility
Checking IPC compatibility with running software
Checking candidate package set infrastructure compatibility
Checking infrastructure compatibility with running software
Checking package specific compatibility
Finished compatibility testing
--- Starting impact testing ---
Checking operational impact of change
Finished impact testing
--- Starting list of software package changes ---

Old files list:
  Removed asr1000rp2-espbase.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-espx86base.03.12.01.S.154-2.S.pkg
New files list:
  Added asr1000rp2-espbase.03.13.00.S.154-3.S-ext.pkg
  Added asr1000rp2-espx86base.03.13.00.S.154-3.S-ext.pkg
  Finished list of software package changes
--- Starting commit of software changes ---
Updating provisioning rollback files
Creating pending provisioning file
Committing provisioning file
Finished commit of software changes
--- Starting analysis of software changes ---
Finished analysis of software changes
--- Starting update running software ---
Blocking peer synchronization of operating information
Creating the command set placeholder directory
  Finding latest command set
  Finding latest command shortlist lookup file
  Finding latest command shortlist file
  Assembling CLI output libraries
  Assembling CLI input libraries
  Assembling Dynamic configuration files
  Applying interim IPC and database definitions
  Replacing running software
  Replacing CLI software
  Restarting software
  Restarting ESP0
  Applying final IPC and database definitions
*Aug  3 14:02:51.450 IST: %IOSXE_OIR-6-OFFLINECARD: Card (fp) offline in slot F0
Generating software version information
Notifying running software of updates
Unblocking peer synchronization of operating information
Unmounting old packages
Cleaning temporary installation files
Finished update running software
SUCCESS: Finished installing software.
```

```
*Aug  3 14:04:49.802 IST: %CPPHA-7-START: F0: cpp_ha:  CPP 0 preparing
image/tmp/sw/fp/0/0/fpx86/mount/usr/cpp/bin/qfp-ucode-esp40
*Aug  3 14:04:50.172 IST: %CPPHA-7-START: F0: cpp_ha:  CPP 0 startup init image
/tmp/sw/fp/0/0/fpx86/mount/usr/cpp/bin/qfp-ucode-esp40
*Aug  3 14:04:50.746 IST: %IOSXE_OIR-6-ONLINECARD: Card (fp) online in slot F0
*Aug  3 14:04:55.480 IST: %CPPHA-7-START: F0: cpp_ha:  CPP 0 running init image
/tmp/sw/fp/0/0/fpx86/mount/usr/cpp/bin/qfp-ucode-esp40
*Aug  3 14:04:55.698 IST: %CPPHA-7-READY: F0: cpp_ha:  CPP 0 loading and initialization
complete
*Aug  3 14:04:55.837 IST: %IOSXE-6-PLATFORM: F0: cpp_cp: Process
CPP_PFILTER_EA_EVENT__API_CALL__REGISTER
```

```
Router# issu commitversion
--- Starting local lock acquisition on R0 ---
Finished local lock acquisition on R0
--- Starting installation changes ---
Cancelling rollback timer
Finished installation changes

SUCCESS: Installation changes committed
```

```
Router# request platform software package install rp 0 file
bootflash:Active_Dir/asr1000rp*03.13.00.S.154-3.S-ext*.pkg
--- Starting local lock acquisition on R0 ---
Finished local lock acquisition on R0
--- Starting file path checking ---
Finished file path checking
--- Starting image file verification ---
Checking image file names
Locating image files and validating name syntax
  Found asr1000rp2-elcbase.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-espbase.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-espx86base.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-rpaccess.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-rpbase.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-rpcontrol.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-rpios-adventerprisek9.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg
Verifying image file locations
Inspecting image file types
    WARNING: In-service installation of RP Base package
    WARNING: requires software reboot of target RP
    WARNING: Automatically setting the on-reboot flag
Processing image file constraints
Creating candidate provisioning file
Finished image file verification
--- Starting candidate package set construction ---
Verifying existing software set
Pocessing candidate provisioning file
Constructing working set for candidate package set
Constructing working set for running package set
Checking command output
Constructing merge of running and candidate packages
Checking if resulting candidate package set would be complete
Finished candidate package set construction
--- Starting compatibility testing ---
Determining whether candidate package set is compatible
```

```
Determining whether installation is valid
Determining whether installation is valid ... skipped
Verifying image type compatibility
Checking IPC compatibility for candidate software
Checking candidate package set infrastructure compatibility
Checking infrastructure compatibility with running software
Checking infrastructure compatibility with running software ... skipped
Checking package specific compatibility
Finished compatibility testing

--- Starting list of software package changes ---

Old files list:
  Removed asr1000rp2-elcbase.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-elcspa.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-rpbase.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-sipbase.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-sipspa.03.12.01.S.154-2.S.pkg

New files list:
  Added asr1000rp2-rpbase.03.13.00.S.154-3.S-ext.pkg
  Finished list of software package changes

--- Starting commit of software changes ---

Updating provisioning rollback files
Creating pending provisioning file
Committing provisioning file
Finished commit of software changes


SUCCESS: Software provisioned.

New software will load on reboot.


Router# show version provisioned
Package: Provisioning File, version: n/a, status: active
  File: bootflash:Active_Dir/packages.conf, on: RP0
  Built: n/a, by: n/a
  File SHA1 checksum: c79075780592aec1312725f4a2357a034fda2d3b

Package: rpbase, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-rpbase.03.13.00.S.154-3.S-ext.pkg, on: RP0
  Built: 2013-07-25_22.55, by: mcpre
  File SHA1 checksum: 4f655c54bb95b4dfa24a0d25ebf97cf8527c69e9

Package: rpcontrol, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-rpcontrol.03.13.00.S.154-3.S-ext.pkg, on: RP0/0
  Built: 2013-07-25_22.55, by: mcpre
  File SHA1 checksum: 8a0a45ea5c7a656c0eef6726174461584f182c78

Package: rpios-adventerprisek9, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-rpios-adventerprisek9.03.13.00.S.154-3.S-ext.pkg,
on: RP0/0
  Built: 2013-07-25_23.00, by: mcpre
  File SHA1 checksum: 85e9eab826bff2194ef568a56c76453625383ad2

Package: rpaccess, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-rpaccess.03.13.00.S.154-3.S-ext.pkg, on: RP0/0
  Built: 2013-07-25_22.55, by: mcpre
  File SHA1 checksum: a360dff0fd76a9b1ae67cda9116c97b62f25ab09

Package: rpcontrol, version: 03.13.00.S.154-3.S-ext, status: n/a
```

```
  File: bootflash:Active_Dir/asr1000rp2-rpcontrol.03.13.00.S.154-3.S-ext.pkg, on: RP0/1
  Built: 2013-07-25_22.55, by: mcpre
  File SHA1 checksum: 8a0a45ea5c7a656c0eef6726174461584f182c78

Package:rpios-adventerprisek9, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-rpios-adventerprisek9.03.13.00.S.154-3.S-ext.pkg,
on: RP0/1
  Built: 2013-07-25_23.00, by: mcpre
  File SHA1 checksum: 85e9eab826bff2194ef568a56c76453625383ad2

Package: rpaccess, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-rpaccess.03.13.00.S.154-3.S-ext.pkg, on: RP0/1
  Built: 2013-07-25_22.55, by: mcpre
  File SHA1 checksum: a360dff0fd76a9b1ae67cda9116c97b62f25ab09

Package: rpbase, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-rpbase.03.13.00.S.154-3.S-ext.pkg, on: RP1
  Built: 2013-07-25_22.55, by: mcpre
  File SHA1 checksum: 4f655c54bb95b4dfa24a0d25ebf97cf8527c69e9

Package: rpcontrol, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-rpcontrol.03.13.00.S.154-3.S-ext.pkg, on: RP1/0
  Built: 2013-07-25_22.55, by: mcpre
  File SHA1 checksum: 8a0a45ea5c7a656c0eef6726174461584f182c78

Package: rpios-adventerprisek9, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-rpios-adventerprisek9.03.13.00.S.154-3.S-ext.pkg,
on: RP1/0
  Built: 2013-07-25_23.00, by: mcpre
  File SHA1 checksum: 85e9eab826bff2194ef568a56c76453625383ad2

Package: rpaccess, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-rpaccess.03.13.00.S.154-3.S-ext.pkg, on: RP1/0
  Built: 2013-07-25_22.55, by: mcpre
  File SHA1 checksum: a360dff0fd76a9b1ae67cda9116c97b62f25ab09

Package: rpcontrol, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-rpcontrol.03.13.00.S.154-3.S-ext.pkg, on: RP1/1
  Built: 2013-07-25_22.55, by: mcpre
  File SHA1 checksum: 8a0a45ea5c7a656c0eef6726174461584f182c78

Package: rpios-adventerprisek9, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-rpios-adventerprisek9.03.13.00.S.154-3.S-ext.pkg,
on: RP1/1
  Built: 2013-07-25_23.00, by: mcpre
  File SHA1 checksum: 85e9eab826bff2194ef568a56c76453625383ad2

Package: rpaccess, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-rpaccess.03.13.00.S.154-3.S-ext.pkg, on: RP1/1
  Built: 2013-07-25_22.55, by: mcpre
  File SHA1 checksum: a360dff0fd76a9b1ae67cda9116c97b62f25ab09

Package: espbase, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-espbase.03.13.00.S.154-3.S-ext.pkg, on: ESP0
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 2fe0ede1545e3f8260b7d453653e812500f0d7b0

Package: espx86base, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-espx86base.03.13.00.S.154-3.S-ext.pkg, on: ESP0
  Built: 2013-07-25_22.55, by: mcpre
  File SHA1 checksum: 571b8bb3866341badd6e24de677b98409f0c789c

Package: espbase, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-espbase.03.13.00.S.154-3.S-ext.pkg, on: ESP1
```

```
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 2fe0ede1545e3f8260b7d453653e812500f0d7b0

Package: spx86base, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-espx86base.03.13.00.S.154-3.S-ext.pkg, on: ESP1
  Built: 2013-07-25_22.55, by: mcpre
  File SHA1 checksum: 571b8bb3866341badd6e24de677b98409f0c789c

Package: sipbase, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg, on: SIP0
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 3b6a4838972840a995ff22e73fd2bae910b268a7

Package: elcbase, version: 03.13.00.S.154-3.S-ext, status: n
  File: bootflash:Active_Dir/asr1000rp2-elcbase.03.13.00.S.154-3.S-ext.pkg, on: SIP0
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 99f8dc925083b118626a4e82d93079050db96826

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP0/0
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP0/0
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP0/1
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP0/1
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP0/2
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP0/2
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP0/3
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP0/3
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipbase, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg, on: SIP1
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 3b6a4838972840a995ff22e73fd2bae910b268a7

Package: elcbase, version: 03.13.00.S.154-3.S-ext, status: n/a
```

```
  File: bootflash:Active_Dir/asr1000rp2-elcbase.03.13.00.S.154-3.S-ext.pkg, on: SIP1
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 99f8dc925083b118626a4e82d93079050db96826

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP1/0
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP1/0
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP1/1
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP1/1
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP1/2
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP1/2
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP1/3
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP1/3
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipbase, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg, on: SIP2
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 3b6a4838972840a995ff22e73fd2bae910b268a7

Package: elcbase, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcbase.03.13.00.S.154-3.S-ext.pkg, on: SIP2
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 99f8dc925083b118626a4e82d93079050db96826

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP2/0
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP2/0
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499
```

```
Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP2/1
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP2/1
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP2/2
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP2/2
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP2/3
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP2/3
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipbase, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg, on: SIP3
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 3b6a4838972840a995ff22e73fd2bae910b268a7

Package: elcbase, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcbase.03.13.00.S.154-3.S-ext.pkg, on: SIP3
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 99f8dc925083b118626a4e82d93079050db96826

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP3/0
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP3/0
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP3/1
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP3/1
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP3/2
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca
```

```
Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP3/2
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP3/3
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP3/3
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipbase, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg, on: SIP4
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 3b6a4838972840a995ff22e73fd2bae910b268a7

Package: elcbase, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcbase.03.13.00.S.154-3.S-ext.pkg, on: SIP4
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 99f8dc925083b118626a4e82d93079050db96826

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP4/0
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP4/0
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP4/1
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP4/1
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP4/2
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP4/2
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP4/3
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP4/3
  Built: 2013-07-25_21.16, by: mcpre
```

```
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipbase, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg, on: SIP5
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 3b6a4838972840a995ff22e73fd2bae910b268a7

Package: elcbase, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcbase.03.13.00.S.154-3.S-ext.pkg, on: SIP5
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 99f8dc925083b118626a4e82d93079050db96826

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP5/0
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP5/0
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP5/1
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP5/1
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP5/2
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP5/2
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP5/3
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP5/3
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499
Router# reload

<some output removed for brevity>


Router# request platform software package clean
Cleaning up unnecessary package files
No path specified, will use booted path bootflash:Active_Dir/packages.conf
Cleaning bootflash:Active_Dir
  Scanning boot directory for packages ... done.
  Preparing packages list to delete ...asr1000rp2-elcbase.03.13.00.S.154-3.S-ext.pkg

      File is in use, will not delete...  asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg
```

```
      File is in use, will not delete...asr1000rp2-espbase.03.13.00.S.154-3.S-ext.pkg
      File is in use, will not delete...asr1000rp2-espx86base.03.13.00.S.154-3.S-ext.pkg
      File is in use, will not delete....asr1000rp2-rpaccess.03.13.00.S.154-3.S-ext.pkg
      File is in use, will not delete...asr1000rp2-rpbase.03.13.00.S.154-3.S-ext.pkg
      File is in use, will not delete...asr1000rp2-rpcontrol.03.13.00.S.154-3.S-ext.pkg
      File is in use, will not
delete...asr1000rp2-rpios-dventerprisek9.03.13.00.S.154-3.S-ext.pkg
      File is in use, will not delete...asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg
      File is in use, will not delete...sr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg
      File is in use, will not delete... packages.conf
      File is in use, will not delete...done.

Files that will be deleted:
  asr1000rp2-elcbase.03.12.01.S.154-2.S.pkg
  asr1000rp2-elcspa.03.12.01.S.154-2.S.pkg
  asr1000rp2-espbase.03.12.01.S.154-2.S.pkg
  asr1000rp2-espx86base.03.12.01.S.154-2.S.pkg
  asr1000rp2-packages-adventerprisek9.03.12.01.S.154-2.S.conf
  asr1000rp2-rpaccess.03.12.01.S.154-2.S.pkg
  asr1000rp2-rpbase.03.12.01.S.154-2.S.pkg
  asr1000rp2-rpcontrol.03.12.01.S.154-2.S.pkg
  asr1000rp2-rpios-adventerprisek9.03.12.01.S.154-2.S.pkg
  asr1000rp2-sipbase.03.12.01.S.154-2.S.pkg
  asr1000rp2-sipspa.03.12.01.S.154-2.S.pkg
  packages.conf.00-
  packages.conf.01-
  packages.conf.02-
  packages.conf.03-
  packages.conf.04-
  packages.conf.05-

Do you want to proceed? [confirm]y
  Deleting file bootflash:Active_Dir/asr1000rp2-elcbase.03.12.01.S.154-2.S.pkg ... done.
  Deleting file bootflash:Active_Dir/asr1000rp2-elcspa.03.12.01.S.154-2.S.pkg ... done.
  Deleting file bootflash:Active_Dir/asr1000rp2-espbase.03.12.01.S.154-2.S.pkg ... done.
  Deleting file bootflash:Active_Dir/asr1000rp2-espx86base.03.12.01.S.154-2.S.pkg ...done.
  Deleting file
bootflash:Active_Dir/asr1000rp2-packages-adventerprisek9.03.12.01.S.154-2.S.conf ... done.
  Deleting file bootflash:Active_Dir/asr1000rp2-rpaccess.03.12.01.S.154-2.S.pkg ... done.
  Deleting file bootflash:Active_Dir/asr1000rp2-rpbase.03.12.01.S.154-2.S.pkg ... done.
  Deleting file bootflash:Active_Dir/asr1000rp2-rpcontrol.03.12.01.S.154-2.S.pkg ... done.
  Deleting file
bootflash:Active_Dir/asr1000rp2-rpios-adventerprisek9.03.12.01.S.154-2.S.pkg ... done.
  Deleting file bootflash:Active_Dir/asr1000rp2-sipbase.03.12.01.S.154-2.S.pkg ... done.
  Deleting file bootflash:Active_Dir/asr1000rp2-sipspa.03.12.01.S.154-2.S.pkg ... done.
  Deleting file bootflash:Active_Dir/packages.conf.00- ... done.
  Deleting file bootflash:Active_Dir/packages.conf.01- ... done.
  Deleting file bootflash:Active_Dir/packages.conf.02- ... done.
  Deleting file bootflash:Active_Dir/packages.conf.03- ... done.
  Deleting file bootflash:Active_Dir/packages.conf.04- ... done.
  Deleting file bootflash:Active_Dir/packages.conf.05- ... done.

SUCCESS: Files deleted.
```

Minimal Disruptive Restart ISSU

A software upgrade of a SIP in any mode results in the SIP being reset and the occurrence of minimal outage during the upgrade. Minimal Disruptive Restart (MDR) minimizes traffic disruption during a software upgrade and supports consolidated package software upgrade and subpackage software upgrade for SIP-40.

The software upgrade of a SIP using MDR is supported only on the Cisco ASR 1000 Series Aggregation Services routers running Cisco IOS XE Release 3.8S and later.

Note MDR upgrades (both consolidated and subpackage upgrades) are supported only on hardware-redundant dual route processor (RP) and Enhanced Services Processors (ESP) platforms.

MDR reduces the downtime (time during which the data plane is unavailable) due to ISSU of a SIP and the SPA(s) within it. The downtime is reduced from 100 seconds to not more than 25 seconds on a SIP reload and from 30 seconds to not more than 10 seconds on a SPA reload. The reload time of a SIP or SPA using a MDR or a non-MDR upgrade remains the same.

MDR for a SIP is permitted if following conditions are met:

- Chassis is hardware redundant (dual RP and dual ESP)
- SIP-type supports MDR.
- At least one MDR-compatible SPA in the SIP should be present for the MDR process.
- All the SPAs associated with the SIP support MDR.
- Both the current software version and the upgrade software version must be ISSU and MDR compatible for both the SIPBase and SIPSPA subpackages for each SPA type that is present.

  The following SPAs support MDR:

  - SPA-2X1GE-V2
  - SPA-5X1GE-V2
  - SPA-8X1GE-V2
  - SPA-10X1GE-V2
  - SPA-1X10GE-L-V2

  Starting with IOS XE release 3.10S and later, MDR support has been extended to include the following SPAs:

  - SPA-2XOC3-POS
  - SPA-4XOC3-POS
  - SPA-4XOC3-POS-V2
  - SPA-8XOC3-POS
  - SPA-1XOC12-POS
  - SPA-2XOC12-POS
  - SPA-4XOC12-POS
  - SPA-8XOC12-POS

Effective from Cisco IOS XE Release 3.12S, the Cisco ASR 1000 Series Fixed Ethernet Line Card (ASR1000-2T+20X1GE) supports Minimal Disruptive Restart (MDR) for ELCBase and ELCSPA packages. MDR support for ASR1000-2T+20X1GE is specifically available on the Cisco ASR 1006 Router and the Cisco ASR 1013 Router with Route Processor 2 (RP2).

For more information on the Cisco ASR 1000 Series Fixed Ethernet Line Card (ASR1000-2T+20X1GE), see the following documents:

Cisco ASR 1000 Series Fixed Ethernet Line Card Hardware Installation Guide

Cisco ASR 1000 Fixed Ethernet Line Card Software Configuration Guide

Use the **request platform software package verify** and **issu checkversion** commands with the **mdr** option to verify the MDR upgrade compatibility of a consolidated package or an individual subpackage.

When performing an ISSU upgrade, you can specify whether the upgrade should use MDR. If you do not specify this, the ISSU upgrade process will remain unchanged.

For more information on ISSU upgrade for nonredundant platforms, see "Upgrade Process with Service Impact for Nonredundant Platforms" section on page 5-74.

Restrictions and Limitations

- MDR is not supported on non-hardware-redundant platforms, such as, Cisco ASR 1001 Router, Cisco ASR 1001-X Router, Cisco ASR 1002 Router, Cisco ASR 1002-X Router, or Cisco ASR 1004 Router, and on hardware-redundant platforms without redundant hardware, such as, Cisco ASR 1006 router, or Cisco ASR 1013 router with only a single RP or ESP.
- MDR is not supported on SIP10.
- MDR does not support software downgrade.
- To use MDR, both the current software and the upgrade software must be ISSU and MDR compatible.
- If a SIP is performing an MDR upgrade the following events cannot be handled during the upgrade:
  - Fast Reroute (FRR)
  - Automatic protection switching (APS)
  - Ethernet over Multiprotocol Label Switching (EoMPLS) and pseudoWire (PW) down that require an attachment circuit (AC) port to be shut.
  - Interface flaps and interface errors
  - Interface-specific or linecard-specific configuration changes
  - SPA Online Insertion and Removal (OIR)
  - Transceiver OIR
  - Line alarms
  - Statistics—Error counts will be reset

High-Availability Considerations

Any high-availability (HA) failover that occurs during MDR-process terminates the process. This includes failures such as active RP IOS failover during the MDR upgrade process or active RP FRU failure in a chassis with redundant hardware during subpackage upgrade.

For MDR upgrade using consolidated packages, upgrade is initiated by the upgraded active RP after the ESP software has been upgraded.

Using ISSU to Perform a Consolidated Package Upgrade in a Dual Route Processor Configuration with MDR

Consolidated packages can only be upgraded using ISSU in dual Route Processor configurations. ISSU is not supported for consolidated package upgrades in single Route Processor configurations.

If you want the RPs on your Cisco ASR 1000 Series router to be running using a consolidated package with MDR after the ISSU upgrade is complete, use the following instructions:

Note This procedure will only work if the current RPs are already running consolidated packages.

SUMMARY STEPS

1. **ip tftp** *source-interface gigabitethernet slot/port*
2. **copy tftp:** *URL-to-target-location*
3. **copy** *source-file-system*:*filename standby-destination-filesystem*
4. **dir** *URL-to-target-location*

   dir *URL-to-target-stby-location*
5. **issu checkversion rp** *upgrade-rp-number* **file** *URL* **mdr {force}**
6. **issu loadversion rp** *upgrade-rp-number* **file** *standby-file-system*:*filename* **mdr {force}**
7. **issu runversion**
8. **telnet** *ip-address port*
9. **issu acceptversion**
10. **issu commitversion**
11. **show version**, **show version** *active-RP* **running**, **show version** *active-RP* **provisioned**

    show platform

    show running-configuration
12. **hw-module slot** *RP slot number* **reload**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**ip tftp** *source-interface gigabitethernet slot/port* **Example:** `Router(config)# ip tftp source-interface gigabitethernet 0`	Specifies the Gigabit Ethernet TFTP source-interface to be configured: *slot/port*—Specifies the location of the TFTP source-interface. **Note** To copy a file using TFTP through the Management Ethernet interface, the **ip tftp source-interface GigabitEthernet 0** command must be entered before entering the **copy tftp** command.
Step 2	**copy tftp:** *URL-to-target-location* **copy** *source-file-system*:*filename standby-destination-filesystem* **Example:** `Router# copy tftp bootflash:`	Copy the consolidated package onto the active RP.
Step 3	**copy** *source-file-system*:*filename standby-destination-filesystem* **Example:** `Router# copy bootflash:asr1000rp2-adventerprisek9.03.13.00.S.154-3.S-ext.bin stby-bootflash:`	Copy the consolidated package onto the standby RP.

Step 4	**dir** *URL-to-target-location*
	dir *URL-to-target-stby-location*
	
	Example:
	`Router# dir bootflash:`
	`Router# dir stby-bootflash:`
		(Optional) Display the contents of the target directories to confirm the successful copy of the file package.

Step 5	**issu checkversion rp** *upgrade-rp-number* **file** *URL* **mdr {force}** **Example:** `Router#issu checkversion rp 1 file stby-bootflash:asr1000rp2-adventerprisek9.03.13.00.S.154-3.S-ext.bin mdr`	Checks the ISSU MDR software packaging compatibility on the standby Route Processor (RP).
Step 6	**issu loadversion rp** *upgrade-rp-number* **file** *standby-file-system*:*filename* **mdr {force}** **Example:** `Router# issu loadversion rp 1 file stby-bootflash:asr1000rp2-adventerprisek9.03.13.00.S.154-3.S-ext.bin mdr`	Load the target consolidated package onto the standby RP. After you receive the message indicating that the terminal state has been reached, go on to Step 7.

	Command or Action	Purpose
Step 7	**issu runversion** **Example:** `Router# issu runversion`	Run the consolidated package that was loaded in Step 6. **Note** If this command is entered before the terminal state is reached, a "`peer is not online`" or "`Standby RP is not in terminal state`" error message will be seen and the **issu runversion** command will not work. If the **issu runversion** command does not run for these reasons, wait for the "`terminal state is reached`" message to appear and retry the **issu runversion** command. You can also monitor the terminal state using the **show platform** command. After ISSU runversion is completed, a switchover will automatically occur and the standby RP will become the active RP.

Step 8	**telnet** *ip-address port*
	
	Example:
	`[unix-server-1 ~]$ telnet 172.17.52.157 2003`
	`User Access Verification`
	
	`Username: user`
	`Password: ********`
	
	`Router>`
		Log in to the RP being upgraded, preferably using the RP's console port, to complete the upgrade. (This is the new active RP, that was the standby RP prior to the ISSU process.)
		
		Note Ensure the hostname does not end in "-stby" after logging into the RP, as this indicates that the RP being accessed is still the standby RP.
		
		There are many ways to log on to a console port. The example shows access to the console port from a UNIX host using telnet.

Step 9	**issu acceptversion** **Example:** `Router# issu acceptversion`	(Optional) Stops the ISSU rollback timer. This step is optional as long as Step 10 is completed before the rollback timer expires.
Step 10	**issu commitversion** **Example:** `Router# issu commitversion`	Completes the ISSU upgrade.

	Command or Action	Purpose
Step 11	**show version** **show version** *active-RP* **running** **show version** *active-RP* **provisioned** **show platform** **show running-configuration**	(Optional) Enter the **show version**, **show platform**, or **show running-configuration** commands to confirm that the upgrade completed successfully, as follows: • **show version**—Confirm that the correct software version is running on your router and that the RP was booted from the correct file. • **show running-configuration**—Confirm that the booting configuration, in particular the boot system statements, are correct. **Tip** Use the **show running-config | include boot** command to display only the boot statements from the **show running-configuration** output. • **show platform**— Confirm that both RP0 and RP1 are correctly running as active and standby.
Step 12	**hw-module slot** *RP-slot* **reload** **Example:** Router# **hw-module slot R0 reload**	Reload the new software on the Standby RP.

Examples

The following example shows how to perform consolidated package upgrade in a dual route processor configuration:

```
Router(config)# ip tftp source-interface gigabitethernet 0
Router#copy tftp: bootflash:
Address or name of remote host []? 172.17.26.81
Source filename []? asr1000rp2-adventerprisek9.03.12.01.S.154-2.S1.bin
Destination filename [asr1000rp2-adventerprisek9.03.12.01.S.154-2.S1.bin]?
Accessing tftp://172.17.26.81/asr1000rp2-adventerprisek9.03.12.01.S.154-2.S1.bin...
Loading asr1000rp2-adventerprisek9.03.12.01.S.154-2.S1.bin from 172.17.26.81 (via
GigabitEthernet0/0/0):
!!!
```

```
Router#copy bootflash: stby-bootflash:
Source filename []? asr1000rp2-adventerprisek9.03.12.01.S.154-2.S1.bin
Destination filename [asr1000rp2-adventerprisek9.03.12.01.S.154-2.S1.bin]?
Copy in
progress...CCCC
```

```
Router#issu checkversion rp 1 file
stby-bootflash:asr1000rp2-adventerprisek9.03.12.01.S.154-2.S1.bin mdr
--- Starting local lock acquisition on R0 ---
Finished local lock acquisition on R0

--- Starting installation state synchronization ---
Finished installation state synchronization

--- Starting local lock acquisition on R1 ---
Finished local lock acquisition on R1

--- Starting file path checking ---
Finished file path checking

--- Starting system installation readiness checking ---
Finished system installation readiness checking

--- Starting image verification ---
Compatibility check with running software on active RP

WARNING:
WARNING: Candidate software combination not found in compatibility database
WARNING:


WARNING:
WARNING: Candidate software combination not found in compatibility database
WARNING:

Software sets are identified as compatible
Finished image verification

--- Starting mdr compatibility verification ---
Extracting consolidated package content
Checking and verifying packages contained in consolidated package
Creating candidate provisioning file
Processing candidate provisioning file
Finished mdr compatibility verification

SUCCESS: Software is ISSU MDR compatible.

Router# issu loadversion rp 1 file
stby-bootflash:asr1000rp2-adventerprisek9.03.12.01.S.154-2.S1.bin mdr
--- Starting local lock acquisition on R0 ---
Finished local lock acquisition on R0

--- Starting installation state synchronization ---
Finished installation state synchronization

--- Starting local lock acquisition on R1 ---
Finished local lock acquisition on R1

--- Starting file path checking ---
Finished file path checking

--- Starting system installation readiness checking ---
Finished system installation readiness checking

--- Starting image verification ---
Compatibility check with running software on active RP

WARNING:
WARNING: Candidate software combination not found in compatibility database
WARNING:
```

```
WARNING:
WARNING: Candidate software combination not found in compatibility database
WARNING:

Software sets are identified as compatible
Finished image verification

--- Starting mdr compatibility verification ---
Extracting consolidated package content
Checking and verifying packages contained in consolidated package
Creating candidate provisioning file
Processing candidate provisioning file
Finished mdr compatibility verification

--- Starting installation changes ---
Setting up image to boot on next reset
Starting automatic rollback timer
Finished installation changes

SUCCESS: Software will now load.

*Mar 14 19:55:44.264: %IOSXE_OIR-6-OFFLINECARD: Card (rp) offline in slot R1
*Mar 14 19:55:44.288: %REDUNDANCY-3-STANDBY_LOST: Standby processor fault
(PEER_NOT_PRESENT)
*Mar 14 19:55:44.288: %REDUNDANCY-3-STANDBY_LOST: Standby processor fault (PEER_DOWN)
*Mar 14 19:55:44.288: %REDUNDANCY-3-STANDBY_LOST: Standby processor fault
(PEER_REDUNDANCY_STATE_CHANGE)
*Mar 14 19:55:46.347: %RF-5-RF_RELOAD: Peer reload. Reason: EHSA standby down
*Mar 14 19:55:46.357: % Redundancy mode change to SSO

Router#
*Mar 14 19:59:37.523: %IOSXE_OIR-6-ONLINECARD: Card (rp) online in slot R1
*Mar 14 19:59:46.889: %REDUNDANCY-5-PEER_MONITOR_EVENT: Active detected a standby
insertion (raw-event=PEER_FOUND(4))

*Mar 14 19:59:46.889: %REDUNDANCY-5-PEER_MONITOR_EVENT: Active detected a standby
insertion (raw-event=PEER_REDUNDANCY_STATE_CHANGE(5))

*Mar 14 19:59:49.794: %REDUNDANCY-3-IPC: IOS versions do not match.

*Mar 14 20:00:58.364: %NBAR_HA-5-NBAR_INFO: NBAR sync DONE!
*Mar 14 20:00:58.777: %HA_CONFIG_SYNC-6-BULK_CFGSYNC_SUCCEED: Bulk Sync succeeded
*Mar 14 20:00:59.778: %RF-5-RF_TERMINAL_STATE: Terminal state reached for (SSO)


Router# issu runversion
--- Starting local lock acquisition on R0 ---
Finished local lock acquisition on R0

--- Starting installation state synchronization ---
Finished installation state synchronization

--- Starting local lock acquisition on R1 ---
Finished local lock acquisition on R1

--- Starting switchover readiness checking ---
Finished switchover readiness checking

SUCCESS: Standby RP will now become active

Initiating active RP failover
```

```
Mar 14 20:02:19.797 R0/0: %PMAN-5-EXITACTION: Process manager is exiting: reload fru
action requested


Initializing Hardware ...

Calculating the ROMMON CRC ... CRC is correct


System Bootstrap, Version 15.2(1r)S, RELEASE SOFTWARE
Copyright (c) 1994-2011 by cisco Systems, Inc.

Current image running: Boot ROM0
Last reset cause: LocalSoft

[Output removed for brevity]



Press RETURN to get started!


*Mar 14 20:08:51.197: %REDUNDANCY-3-PEER_MONITOR: PEER_FOUND event on standby
*Mar 14 20:09:53.509: %SYS-5-RESTART: System restarted --
Cisco IOS Software, IOS-XE Software (X86_64_LINUX_IOSD-ADVENTERPRISEK9-M), Version
15.3(1)S, RELEASE SOFTWARE (fc4)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2012 by Cisco Systems, Inc.
Compiled Tue 27-Nov-12 11:13 by mcpre
*Mar 14 20:10:00.174: %PLATFORM-6-RF_PROG_SUCCESS: RF state STANDBY HOT
Router-stby>
```

At this point of the procedure, use your UNIX client to log in to the other RP:

```
[unix-server-1 ~]$ telnet 172.17.152.157 2013
User Access Verification

Username: user
Password: ********

Router>
*Mar 14 20:02:15.767: %LINK-3-UPDOWN: Interface Lsmpi0, changed state to up
*Mar 14 20:02:15.768: %LINK-3-UPDOWN: Interface EOBC0, changed state to up
*Mar 14 20:02:15.768: %LINK-3-UPDOWN: Interface LIIN0, changed state to up
*Mar 14 20:02:16.768: %LINEPROTO-5-UPDOWN: Line protocol on Interface Lsmpi0, changed
state to up
*Mar 14 20:02:16.768: %LINEPROTO-5-UPDOWN: Line protocol on Interface EOBC0, changed state
to up
*Mar 14 20:02:16.768: %LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0,
changed state to down
*Mar 14 20:02:16.768: %LINEPROTO-5-UPDOWN: Line protocol on Interface LIIN0, changed state
to up
*Mar 14 20:02:17.498: %CRYPTO-6-ISAKMP_ON_OFF: ISAKMP is OFF
*Mar 14 20:02:17.498: %CRYPTO-6-GDOI_ON_OFF: GDOI is OFF
*Mar 14 20:02:17.756: %LINK-3-UPDOWN: Interface Null0, changed state to up
*Mar 14 20:02:17.757: %LINK-3-UPDOWN: Interface GigabitEthernet0/0/0, changed state to up
*Mar 14 20:02:17.757: %LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback0, changed
state to up
*Mar 14 20:02:17.757: %LINK-3-UPDOWN: Interface Loopback0, changed state to up
*Mar 14 20:02:17.765: %LINK-5-CHANGED: Interface GigabitEthernet0/0/1, changed state to
administratively down
*Mar 14 20:02:17.765: %LINK-5-CHANGED: Interface GigabitEthernet0/0/2, changed state to
administratively down
```

```
*Mar 14 20:02:17.765: %LINK-5-CHANGED: Interface GigabitEthernet0/0/3, changed state to
administratively down
*Mar 14 20:02:17.765: %LINK-5-CHANGED: Interface GigabitEthernet0/0/4, changed state to
administratively down
*Mar 14 20:02:17.765: %LINK-5-CHANGED: Interface GigabitEthernet0/0/5, changed state to
administratively down
*Mar 14 20:02:17.765: %LINK-5-CHANGED: Interface GigabitEthernet0/0/6, changed state to
administratively down
*Mar 14 20:02:17.765: %LINK-5-CHANGED: Interface GigabitEthernet0/0/7, changed state to
administratively down
*Mar 14 20:02:17.765: %LINK-5-CHANGED: Interface GigabitEthernet0/1/0, changed state to
administratively down
*Mar 14 20:02:17.765: %LINK-5-CHANGED: Interface GigabitEthernet0/1/1, changed state to
administratively down
*Mar 14 20:02:18.756: %LINEPROTO-5-UPDOWN: Line protocol on Interface Null0, changed state
to up
*Mar 14 20:02:18.757: %LINEPROTO-5-UPDOWN: Line protocol on Interface
GigabitEthernet0/0/0, changed state to up
*Mar 14 20:04:05.534: %CPPHA-7-START: F1: cpp_ha: CPP 0 preparing image
/tmp/sw/fp/1/0/fpx86/mount/usr/cpp/bin/qfp-ucode-esp40
*Mar 14 20:04:05.863: %CPPHA-7-START: F1: cpp_ha: CPP 0 startup init image
/tmp/sw/fp/1/0/fpx86/mount/usr/cpp/bin/qfp-ucode-esp40
*Mar 14 20:04:07.781: %IOSXE_OIR-6-ONLINECARD: Card (fp) online in slot F1
*Mar 14 20:04:11.123: %CPPHA-7-START: F1: cpp_ha: CPP 0 running init image
/tmp/sw/fp/1/0/fpx86/mount/usr/cpp/bin/qfp-ucode-esp40
*Mar 14 20:04:11.342: %CPPHA-7-READY: F1: cpp_ha: CPP 0 loading and initialization
complete
*Mar 14 20:04:11.472: %IOSXE-6-PLATFORM: F1: cpp_cp: Process
CPP_PFILTER_EA_EVENT__API_CALL__REGISTER
*Mar 14 20:04:19.443: %PLATFORM-5-RESETCARD: R1/0: card_reset: ESP0 reset to acquire
provisioned software
*Mar 14 20:04:19.458: %IOSXE_OIR-6-OFFLINECARD: Card (fp) offline in slot F0
*Mar 14 20:04:21.546: %MDR-5-CARD_RESTART: R1/0: card_mdr: Minimal Disruptive Restart SIP0
to acquire provisioned software
*Mar 14 20:04:29.266: %CMCC-5-SPA_MDR_INIT: SIP0: cmcc: SPA0 initiated Minimal Disruptive
Restart
*Mar 14 20:04:29.269: %CMCC-5-SPA_MDR_INIT: SIP0: cmcc: SPA1 initiated Minimal Disruptive
Restart
*Mar 14 20:04:29.270: %CMCC-5-SPA_MDR_INIT: SIP0: cmcc: SPA2 initiated Minimal Disruptive
Restart
*Mar 14 20:05:17.807: %IOSXE_OIR-6-ONLINECARD: Card (cc) online in slot 0
*Mar 14 20:05:21.287: %IOSXE_OIR-6-INSSPA: SPA inserted in subslot 0/0
*Mar 14 20:05:21.630: %IOSXE_OIR-6-INSSPA: SPA inserted in subslot 0/1
*Mar 14 20:05:21.896: %IOSXE_OIR-6-INSSPA: SPA inserted in subslot 0/2
*Mar 14 20:05:29.364: %LINK-3-UPDOWN: SIP0/0: Interface EOBC0/1, changed state to up
*Mar 14 20:05:34.058: %SPA_OIR-6-ONLINECARD: SPA (SPA-8X1GE-V2) online in subslot 0/0
*Mar 14 20:05:33.064: %CMCC-5-SPA_MDR_DONE: SIP0: cmcc: SPA0 completed Minimal Disruptive
Restart
*Mar 14 20:05:34.093: %LINK-3-UPDOWN: SIP0/1: Interface EOBC0/1, changed state to up
*Mar 14 20:05:35.125: %SPA_OIR-6-ONLINECARD: SPA (SPA-8X1GE-V2) online in subslot 0/1
*Mar 14 20:05:34.130: %CMCC-5-SPA_MDR_DONE: SIP0: cmcc: SPA1 completed Minimal Disruptive
Restart
*Mar 14 20:05:34.099: %TRANSCEIVER-6-INSERTED: SIP0/1: transceiver module inserted in
GigabitEthernet0/1/0
*Mar 14 20:05:35.046: %LINK-3-UPDOWN: SIP0/0: Interface GigabitEthernet0/0/0, changed
state to up
*Mar 14 20:05:37.996: %LINK-3-UPDOWN: SIP0/2: Interface EOBC0/1, changed state to up
*Mar 14 20:05:43.230: %SPA_OIR-6-ONLINECARD: SPA (SPA-8X1GE-V2) online in subslot 0/2
*Mar 14 20:05:42.239: %CMCC-5-SPA_MDR_DONE: SIP0: cmcc: SPA2 completed Minimal Disruptive
Restart
*Mar 14 20:06:10.368: %CPPHA-7-START: F0: cpp_ha: CPP 0 preparing image
/tmp/sw/fp/0/0/fpx86/mount/usr/cpp/bin/qfp-ucode-esp40
*Mar 14 20:06:10.697: %CPPHA-7-START: F0: cpp_ha: CPP 0 startup init image
/tmp/sw/fp/0/0/fpx86/mount/usr/cpp/bin/qfp-ucode-esp40
```

```
*Mar 14 20:06:12.158: %IOSXE_OIR-6-ONLINECARD: Card (fp) online in slot F0
*Mar 14 20:06:15.940: %CPPHA-7-START: F0: cpp_ha: CPP 0 running init image
/tmp/sw/fp/0/0/fpx86/mount/usr/cpp/bin/qfp-ucode-esp40
*Mar 14 20:06:16.158: %CPPHA-7-READY: F0: cpp_ha: CPP 0 loading and initialization
complete
*Mar 14 20:06:16.290: %IOSXE-6-PLATFORM: F0: cpp_cp: Process
CPP_PFILTER_EA_EVENT__API_CALL__REGISTER
*Mar 14 20:08:41.564: %IOSXE_OIR-6-ONLINECARD: Card (rp) online in slot R0
*Mar 14 20:08:51.188: %REDUNDANCY-5-PEER_MONITOR_EVENT: Active detected a standby
insertion (raw-event=PEER_FOUND(4))

*Mar 14 20:08:51.188: %REDUNDANCY-5-PEER_MONITOR_EVENT: Active detected a standby
insertion (raw-event=PEER_REDUNDANCY_STATE_CHANGE(5))

*Mar 14 20:08:54.115: %REDUNDANCY-3-IPC: IOS versions do not match.

*Mar 14 20:10:00.611: %NBAR_HA-5-NBAR_INFO: NBAR sync DONE!
*Mar 14 20:10:01.026: %HA_CONFIG_SYNC-6-BULK_CFGSYNC_SUCCEED: Bulk Sync succeeded
*Mar 14 20:10:02.027: %RF-5-RF_TERMINAL_STATE: Terminal state reached for (SSO)
Router>enable



Router# issu acceptversion
--- Starting local lock acquisition on R1 ---
Finished local lock acquisition on R1

Cancelling rollback timer
SUCCESS: Rollback timer cancelled


Router# issu commitversion
--- Starting local lock acquisition on R1 ---
Finished local lock acquisition on R1

--- Starting commit readiness checking ---
Finished commit readiness checking

--- Starting installation changes ---
Cancelling rollback timer
Saving image changes
Finished installation changes

Building configuration...
[OK]
SUCCESS: version committed: bootflash:asr1000rp2-adventerprisek9.03.12.01.S.154-2.S1.bin
```

After you confirm the software versions and configuration, reload the RP as shown in the following example:

```
Router# hw-module slot R0 reload
  Proceed with reload of module? [confirm]
*Mar 14 20:31:05.511: %IOSXE_OIR-6-OFFLINECARD: Card (rp) offline in slot R0
*Mar 14 20:31:05.587: %REDUNDANCY-3-STANDBY_LOST: Standby processor fault
(PEER_NOT_PRESENT)
*Mar 14 20:31:05.588: %REDUNDANCY-3-STANDBY_LOST: Standby processor fault (PEER_DOWN)
*Mar 14 20:31:05.588: %REDUNDANCY-3-STANDBY_LOST: Standby processor fault
(PEER_REDUNDANCY_STATE_CHANGE)
*Mar 14 20:31:07.622: %RF-5-RF_RELOAD: Peer reload. Reason: EHSA standby down
*Mar 14 20:31:07.632: % Redundancy mode change to SSO

*Mar 14 20:34:29.337: %IOSXE_OIR-6-ONLINECARD: Card (rp) online in slot R0
*Mar 14 20:34:38.988: %REDUNDANCY-5-PEER_MONITOR_EVENT: Active detected a standby
insertion (raw-event=PEER_FOUND(4))
```

```
*Mar 14 20:34:38.988: %REDUNDANCY-5-PEER_MONITOR_EVENT: Active detected a standby
insertion (raw-event=PEER_REDUNDANCY_STATE_CHANGE(5))

*Mar 14 20:35:47.610: %NBAR_HA-5-NBAR_INFO: NBAR sync DONE!
*Mar 14 20:35:48.023: %HA_CONFIG_SYNC-6-BULK_CFGSYNC_SUCCEED: Bulk Sync succeeded
*Mar 14 20:35:49.025: %RF-5-RF_TERMINAL_STATE: Terminal state reached for (SSO)
```

Using ISSU to Upgrade the Subpackages in a Dual Route Processor Configuration with MDR

This section provides instructions on performing an ISSU upgrade on a Cisco ASR 1000 Series Router with dual RPs that are currently running individual subpackages.

This section covers the following topics:

- Using ISSU to Upgrade the Subpackages on a Cisco ASR 1006 Router or Cisco ASR 1013 Router (issu Command Set) with MDR, page 5-164
- Using ISSU to Upgrade Subpackages on a Cisco ASR 1006 Router or Cisco ASR 1013 Router (request platform command set) with MDR, page 5-206

Using ISSU to Upgrade the Subpackages on a Cisco ASR 1006 Router or Cisco ASR 1013 Router (issu Command Set) with MDR

This section provides the instructions for performing an ISSU upgrade using subpackages with MDR on a Cisco ASR 1006 router or ASR 1013 router with a dual RP setup using the **issu** command set.

This procedure can only be performed if the current ASR 1006 or ASR 1013 routers have two active RPs and both RPs are running subpackages.

SUMMARY STEPS

1. **show version**

   show version *active-rp* **installed**

   dir *filesystem*:<*directory*>

   show platform

   show redundancy-states
2. **copy running-config startup-config**
3. **mkdir** *URL-to-directory-name*
4. **ip tftp** *source-interface gigabitethernet port*
5. **copy tftp:** *URL-to-target-location*
6. **request platform software package expand file** *URL-to-consolidated-package*
7. **dir** *URL-to-consolidated-package*
8. **copy** *file-system*:**asr1000rp2-espbase.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*

copy *file-system*:**asr1000rp2-espx86base.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*

copy *file-system*:**asr1000rp2-rpaccess.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*

copy *file-system*:**asr1000rp2-rpbase.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*

copy *file-system*:**asr1000rp2-rpcontrol.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*

copy *file-system*:**asr1000rp2-rpios.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*

copy *file-system*:**asr1000rp2-sipbase.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*

copy *file-system*:**asr1000rp2-sipspa.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*

copy *file-system*:**asr1000rp2-elcbase.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*

copy *file-system*:**asr1000rp2-elcspa.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*

Note In step 9, each individual subpackage that was extracted in step 6 is copied to the directory where the subpackages that are currently running the standby RP are stored.

9. **copy** *file-system*:**asr1000rp2-espbase.***version*.**pkg** *URL-to-directory-of-sub-packages-standby-RP*

   copy *file-system*:**asr1000rp2-espx86base.***version*.**pkg** *URL-to-directory-of-sub-packages-standby-RP*

   copy *file-system*:**asr1000rp2-rpaccess.***version*.**pkg** *URL-to-directory-of-sub-packages-standby-RP*

   copy *file-system*:**asr1000rp2-rpbase.***version*.**pkg** *URL-to-directory-of-sub-packages-standby-RP*

   copy *file-system*:**asr1000rp2-rpcontrol.***version*.**pkg** *URL-to-directory-of-sub-packages-standby-RP*

   copy *file-system*:**asr1000rp2-rpios.***version*.**pkg** *URL-to-directory-of-sub-packages-standby-RP*

   copy *file-system*:**asr1000rp2-sipbase.***version*.**pkg** *URL-to-directory-of-sub-packages-standby-RP*

   copy *file-system*:**asr1000rp2-sipspa.***version*.**pkg** *URL-to-directory-of-sub-packages-standby-RP*

   copy *file-system*:**asr1000rp2-elcbase.***version*.**pkg** *URL-to-directory-of-sub-packages-standby-RP*

   copy *file-system*:**asr1000rp2-elcspa.***version*.**pkg** *URL-to-directory-of-sub-packages-standby-RP*

10. **issu checkversion rp** *standby-RP* **file** *standy-file-system*:**asr1000rp****version**.**pkg mdr {force}**

11. **issu loadversion rp** *standby-RP* **file** *URL-to-standby-file-system*:**asr1000rp****version**.**pkg force**

12. **hw-module slot** *standby-RP* **reload**

13. **issu loadversion rp** *active-RP* **file** *URL-to-active-file-system*:**asr1000rp1-{sipbase,sipspa}****version**.**pkg slot** *SIP-slot-number* **mdr {force}**

    issu commitversion

    Note Repeat this step for each SIP installed in the router before moving to the next step.

14. **issu loadversion rp** *active-RP* **file** *URL-to-active-file-system*:**asr1000rp2-{elcbase,elcspa}****version**.**pkg slot** *ELC-slot-number* **mdr {force}**

    issu commitversion

Note Repeat this step for each ELC installed in the router before moving to the next step.

15. **issu loadversion rp** *active-RP* **file** *URL-to-active-file-system*:**asr1000rp2-esp***version***.pkg slot** *standby-ESP-slot*

    issu commitversion

    issu loadversion rp *active-RP* **file** *URL-to-active-file-system*:**asr1000rp2-esp***version***.pkg slot** *active-ESP-slot*

    issu commitversion

16. **issu loadversion rp** *active-RP* **file** *URL-to-active-file-system*:**asr1000rp2***version***.pkg force**
17. **show version** *active-RP* **provisioned**

    show version *active-RP* **installed**

18. **redundancy force-switchover**
19. **request platform software package clean**

DETAILED STEPS

	Command or Action	Purpose

Step 1	**show version**
	show version *active-rp* **installed**
	dir *filesystem*: *directory*
	show platform
	show redundancy states
	
	Example:
	`Router# show version`
	`Router# show version r0 installed`
	`Router# dir bootflash:`
	`Router# show platform`
	`Router# show redundancy states`
		(Optional) Use the following commands to confirm the current router configuration, as follows:
		• **show version** and **show version** *active-rp* **installed**—Verify the running version of the Cisco IOS XE software on the router, and which file was used to boot the router, and where that file is stored.
		• **dir**—Confirm that the files that were used to boot the router are located in the directory.
		• **show platform**—Confirm the current status of the active and standby RPs.
		• **show redundancy states**—Confirm the operational and configured redundancy states.

Step 2	**copy running-config startup-config** **Example:** `Router# copy running-config startup-config`	After you have confirmed that the system states are acceptable, save the current configuration to the startup configuration.
Step 3	**mkdir** *URL-to-directory-name* **Example:** `Router# mkdir bootflash:tmp`	Create a directory to store the consolidated package and subpackages. This directory must be created in most cases because the consolidated packages and subpackages have to be separated from the subpackages that booted the router at this step of the procedure.
Step 4	**ip tftp source-interface gigabitethernet** *port* **Example:** `Router(config)# ip tftp source-interface gigabitethernet 0`	Specifies the Gigabit Ethernet TFTP source-interface to be configured: *slot/port*—Specifies the location of the TFTP source-interface. **Note** To copy a file using TFTP through the Management Ethernet interface, the **ip tftp source-interface GigabitEthernet 0** command must be entered before entering the **copy tftp** command.
Step 5	**copy tftp:** *URL-to-target-location* **Example:** `Router# copy tftp: bootflash:tmp`	Copy the consolidated package file into the directory created in Step 3. The consolidated package in this step should not be copied into the same directory where the subpackages that are currently running your router are stored (the directory containing the packages.conf provisioning file from which the router was booted). **Tip** It is recommended that you copy the package onto a usb: or harddisk: file system for space considerations when performing this step of the procedure.

	Command or Action	Purpose
Step 6	**request platform software package expand file** *URL-to-consolidated-package* **Example:** `Router# request platform software package expand file bootflash:tmp/asr1000rp2-adventerprisek.9.03.13.00.S.154-3.S-ext.bin`	Extract the subpackages out of the consolidated package file into the temporary directory. **Note** Take extra care to extract the subpackages to a temporary subdirectory and do not delete any of the files currently running the router at this point of the procedure. To erase the files that were running on the router before the ISSU upgrade, enter the **request platform software package clean** command after the ISSU upgrade has been completed.
Step 7	**dir** *target-URL* **Example:** `Router# dir bootflash:tmp`	(Optional) Display the directory to confirm that the files were extracted.

Command or Action	Purpose

Step 8 **copy** *file-system*:**asr1000rp2-espbase.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*
copy *file-system*:**asr1000rp2-espx86base.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*
copy *file-system*:**asr1000rp2-rpaccess.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*
copy *file-system*:**asr1000rp2-rpbase.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*
copy *file-system*:**asr1000rp2-rpcontrol.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*
copy *file-system*:**asr1000rp2-rpios.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*
copy *file-system*:**asr1000rp2-sipbase.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*
copy *file-system*:**asr1000rp2-sipspa.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*
copy *file-system*:**asr1000rp2-elcbase.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*
copy *file-system*:**asr1000rp2-elcspa.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*
 	
Example:
`Router# copy bootflash:tmp/asr1000rp2-espbase.03.13.00.S.154-3.S-ext.pkg bootflash:`
 	
`Router# copy bootflash:tmp/asr1000rp2-espx86base.03.13.00.S.154-3.S-ext.pkg bootflash:`
 	
`Router# copy bootflash:tmp/asr1000rp2-rpaccess.03.13.00.S.154-3.S-ext.pkg bootflash:`
 	
`Router# copy bootflash:tmp/asr1000rp2-rpbase.03.13.00.S.154-3.S-ext.pkg bootflash:`
 	
`Router# copy bootflash:tmp/asr1000rp2-rpcontrol.03.13.00.S.154-3.S-ext.pkg bootflash:`
 	
`Router# copy bootflash:tmp/asr1000rp2-rpios-adventerprisek9.03.13.00.S.154-3.S-ext.pkg bootflash:`
 	
`Router# copy bootflash:tmp/asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg bootflash:`
 	
`Router# copy bootflash:tmp/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg bootflash:`
 	
`Router# copy bootflash:tmp/asr1000rp2-elcbase.03.13.00.S.154-3.S-ext.pkg bootflash:`
 	
`Router# copy bootflash:tmp/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg bootflash:`
	Copy the subpackages out of the temporary directory into the directory on the router where the subpackages running the active RP are currently stored.

Command or Action	Purpose
Step 9 **copy** *file-system*:**asr1000rp2-espbase.**_version_.**pkg** *URL-to-directory-of-sub-packages-standby-RP* **copy** *file-system*:**asr1000rp2-espx86base.**_version_.**pkg** *URL-to-directory-of-sub-packages-standby-RP* **copy** *file-system*:**asr1000rp2-rpaccess.**_version_.**pkg** *URL-to-directory-of-sub-packages-standby-RP* **copy** *file-system*:**asr1000rp2-rpbase.**_version_.**pkg** *URL-to-directory-of-sub-packages-standby-RP* **copy** *file-system*:**asr1000rp2-rpcontrol.**_version_.**pkg** *URL-to-directory-of-sub-packages-standby-RP* **copy** *file-system*:**asr1000rp2-rpios.**_version_.**pkg** *URL-to-directory-of-sub-packages-standby-RP* **copy** *file-system*:**asr1000rp2-sipbase.**_version_.**pkg** *URL-to-directory-of-sub-packages-standby-RP* **copy** *file-system*:**asr1000rp2-sipspa.**_version_.**pkg** *URL-to-directory-of-sub-packages-standby-RP* **copy** *file-system*:**asr1000rp2-elcbase.**_version_.**pkg** *URL-to-directory-of-sub-packages-standby-RP* **copy** *file-system*:**asr1000rp2-elcspa.**_version_.**pkg** *URL-to-directory-of-sub-packages-standby-RP* **Examples:** Router# **copy bootflash:tmp/asr1000rp2-espbase.03.13.00.S.154-3.S-ext.pkg stby-bootflash:** Router# **copy bootflash:tmp/asr1000rp2-espx86base.03.13.00.S.154-3.S-ext.pkg stby-bootflash:** Router# **copy bootflash:tmp/asr1000rp2-rpaccess.03.13.00.S.154-3.S-ext.pkg stby-bootflash:** Router# **copy bootflash:tmp/asr1000rp2-rpbase.03.13.00.S.154-3.S-ext.pkg stby-bootflash:** Router# **copy bootflash:tmp/asr1000rp2-rpcontrol.03.13.00.S.154-3.S-ext.pkg stby-bootflash:** Router# **copy bootflash:tmp/asr1000rp2-rpios-adventerprisek9.03.13.00.S.154-3.S-ext.pkg** Router# **copy bootflash:tmp/asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg stby-bootflash:** Router# **ccopy bootflash:tmp/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg stby-bootflash:** Router# **copy bootflash:tmp/asr1000rp2-elcbase.03.13.00.S.154-3.S-ext.pkg stby-bootflash:** Router# **ccopy bootflash:tmp/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg stby-bootflash:**	Copy the subpackages out of the temporary directory into the directory on the router where the subpackages running the standby RP are currently stored.

	Command or Action	Purpose
Step 10	**issu checkversion rp** *standby-RP* **file** *standy-file-system*:**asr1000rp***version***.pkg mdr {force}** **Example:** `Router#issu checkversion rp 1 file stby-bootflash:asr1000rp2-*03.13.00.S.154-3.S-ext*pkg mdr`	Checks the ISSU MDR software packaging compatibility on the standby Route Processor (RP).
Step 11	**issu loadversion rp** *standby-RP* **file** *target-standbyRP-URL-for-sub-packages*:**asr1000rp*** *version****.pkg force** **Example:** `Router# issu loadversion rp 1 file stby-bootflash:asr1000rp2-*03.13.00.S.154-3.S-ext*pkg force`	Upgrade the RP subpackages on the standby RP, where the "**rp***" wildcard is specified to capture all of the RP subpackages for the desired upgrade release.
Step 12	**hw-module slot** *standby-RP* **reload** **Example:** `Router# hw-module slot R1 reload`	Reload the standby RP.

Step 13	**issu loadversion rp** *active-RP* **file** *URL-to-active-file-system*:**asr1000rp2-{sipbase,sipspa}***version****.pkg slot** *SIP-slot-number* **mdr {force}**
	issu commitversion
	
	Example:
	`Router# issu loadversion rp 0 file bootflash:asr1000rp2-{sipbase,sipspa}*03.13.00*pkg mdr`
	`Router# issu commitversion`
		Upgrade the SIP and SPA subpackages for each SIP on the router using MDR.
		
		Note This step must be completed one SIP at a time, and repeated for each SIP installed on the router before performing the next step.
		
		Tip You can use the **show ip interface brief** command to identify which slots contain SIPs and SPAs. The interfaces with three numbers (in the form *SIP-number*/*SPA-number*/*interface-number*) identify the SIP and SPA locations in the router.
		
		Note The *pattern* options used in this CLI (**sipbase** and **sipspa**) were introduced in Cisco IOS XE Release 2.1.2 and are not available in previous Cisco IOS XE Releases.
		See the "ISSU Procedures (Prior to Cisco IOS XE Release 2.1.2)" section on page 5-72 for pre-Cisco IOS XE Release 2.1.2 ISSU upgrade procedures.

	Command or Action	Purpose

Step 14	**issu loadversion rp** *active-RP* **file** *URL-to-active-file-system*:**asr1000rp2-{elcbase,elcspa}***version****.pkg slot** *ELC-slot-number* **mdr {force}**
	issu commitversion
	
	Example:
	`Router# issu loadversion rp 0 file bootflash:asr1000rp2-{elcbase,elcspa}*03.13.00.S.154-3.S-ext*pkg mdr`
	`Router# issu commitversion`
		Upgrade the Ethernet Line Card (ELC) and SPA subpackages for each ELC on the router using MDR.
		Note This step must be completed one ELC at a time, and repeated for each ELC installed on the router before performing the next step.
		Tip You can use the **show ip interface brief** command to identify which slots contain ELCs and SPAs. The interfaces with three numbers (in the form *ELC-number*/*SPA-number*/*interface-number*) identify the ELC and SPA locations in the router.
		Note The *pattern* options used in this CLI (**elcbase** and **elcspa**) were introduced in Cisco IOS XE Release 3.10S and are not available in previous Cisco IOS XE Releases.

Step 15	**issu loadversion rp** *active-RP* **file** *URL-to-active-file-system*:**asr1000rp2-esp***version****.pkg slot** *standby*-ESP-*slot*
	issu commitversion
	issu loadversion rp *active-RP* **file** *URL-to-active-file-system*:**asr1000rp2-esp***version****.pkg slot** *active*-ESP-*slot*
	issu commitversion
	
	Example:
	`Router# issu loadversion rp 0 file bootflash:asr1000rp2-esp*03.13.00.S.154-3.S-ext*pkg slot 1`
	`Router# issu commitversion`
	
	`Router# issu loadversion rp 0 file bootflash:asr1000rp2-esp*03.13.00.S.154-3.S-ext*pkg slot 0`
	`Router# issu commitversion`
		Upgrade the ESP Base subpackage on the standby and the active ESPs.
		After entering the **issu loadversion rp** command on the active RP, the ESP switchover will occur automatically. Minimal traffic interruption will occur as a result of this switchover.

Step 16	**issu loadversion rp** *active-RP* **file** *URL-to-active-file-system*:**asr1000rp***version****.pkg force** **Example:** `Router# issu loadversion rp 0 file bootflash:asr1000rp2*03.13.00.S.154-3.S-ext*pkg force`	Upgrade all of the subpackages on the active RP. **Note** This step is required to ensure that all subpackages on the router were upgraded as part of this procedure, and might upgrade some subpackages that would otherwise be missed in the process.

Step 17	**show version** *active-RP* **provisioned**
	show version *active-RP* **installed**
	
	Example:
	`Router# show version r0 provisioned`
	`Router# show version r0 installed`
		(Optional) Confirm that the subpackages are provisioned and installed.

	Command or Action	Purpose
Step 18	**redundancy force-switchover** **Example:** `Router# redundancy force-switchover`	Force an RP switchover to complete the upgrade.
Step 19	**request platform software package clean** **Example:** `Router# request platform software package clean`	(Optional) Removes all unused subpackages files from the router.

Examples

The following example shows ISSU upgrade using subpackages on a Cisco ASR 1006 router or ASR 1013 router with a dual RP setup using the **issu** command set

```
Router# show version
Cisco IOS Software, IOS-XE Software (X86_64_LINUX_IOSD-ADVENTERPRISEK9-M), Version
15.3(2)S, RELEASE SOFTWARE (fc1)

<output removed for brevity>

System image file is "bootflash:Active_Dir/packages.conf"

<output removed for brevity>
cisco ASR1013 (RP2) processor with 4208889K/6147K bytes of memory.
Processor board ID FOX1343GJGC
20 Gigabit Ethernet interfaces
6 Ten Gigabit Ethernet interfaces
32768K bytes of non-volatile configuration memory.
8388608K bytes of physical memory.
1925119K bytes of eUSB flash at bootflash:.
78085207K bytes of SATA hard disk at harddisk:.

Configuration register is 0x2102

Router# show platform
Chassis type: ASR1013

Slot       Type                State                 Insert time (ago)
---------  ------------------- --------------------- -----------------
 2         ASR1000-SIP40       ok                    1d03h
 2/0       SPA-1X10GE-L-V2     ok                    1d03h
 2/1       SPA-1X10GE-L-V2     ok                    1d03h
 2/2       SPA-1X10GE-L-V2     ok                    1d03h
 2/3       SPA-1X10GE-L-V2     ok                    1d03h
 4         ASR1000-2T+20X1GE   ok                    1d03h
 4/0       BUILT-IN-2T+20X1GE  ok                    1d03h
 R0        ASR1000-RP2         ok, active            1d03h
 R1        ASR1000-RP2         ok, standby           1d03h
 F0        ASR1000-ESP100      ok, active            1d03h
 F1        ASR1000-ESP100      ok, standby           1d03h
 P0        ASR1013-PWR-AC      ok                    1d03h
 P1        ASR1013-PWR-AC      ok                    1d03h
 P2        ASR1013-PWR-AC      ok                    1d03h
 P3        ASR1013-PWR-AC      ps, fail              1d03h

Slot      CPLD Version        Firmware Version
--------- ------------------- ----------------------------------------
2         00200800            15.3(3r)S
4         00200800            15.3(1r)S
```

```
R0        10021901            15.3(3r)S
R1        10021901            15.3(3r)S
F0        12071700            15.3(3r)S
F1        12071700            15.3(3r)S

Router# show version r0 installed
Package: Provisioning File, version: n/a, status: active
  File: bootflash:Active_Dir/packages.conf, on: RP0
  Built: n/a, by: n/a
  File SHA1 checksum: a624f70f68c60292f4482433f43afd92487a55c4

Package: rpbase, version: 03.12.01.S.154-2.S, status: active
  File: bootflash:Active_Dir/asr1000rp2-rpbase.03.12.01.S.154-2.S.pkg, on: RP0
  Built: 2013-03-25_18.48, by: mcpre
  File SHA1 checksum: 3a9675142898cfac350d4e42f0e37bd9f4e48538

Package: rpcontrol, version: 03.12.01.S.154-2.S, status: active
  File: bootflash:Active_Dir/asr1000rp2-rpcontrol.03.12.01.S.154-2.S.pkg, on: RP0/0
  Built: 2013-03-25_18.48, by: mcpre
  File SHA1 checksum: 87b11f863f67fdf2610ee0769b929baab4c3efad

<output removed for brevity>

Router# dir bootflash:Active_Dir
Directory of bootflash:/Active_Dir/

   20  -rw-    41104112   Aug 3 2013 15:05:40 +05:30
asr1000rp2-elcbase.03.12.01.S.154-2.S.pkg
   21  -rw-    50285296   Aug 3 2013 15:05:40 +05:30
asr1000rp2-elcspa.03.12.01.S.154-2.S.pkg
   22  -rw-    82514676   Aug 3 2013 15:05:40 +05:30
asr1000rp2-espbase.03.12.01.S.154-2.S.pkg
   23  -rw-   101084628   Aug 3 2013 15:05:40 +05:30
asr1000rp2-espx86base.03.12.01.S.154-2.S.pkg
   17  -rw-        9059   Aug 3 2013 15:05:40 +05:30
asr1000rp2-packages-adventerprisek9.03.12.01.S.154-2.S.conf
   24  -rw-    29012724   Aug 3 2013 15:05:40 +05:30
asr1000rp2-rpaccess.03.12.01.S.154-2.S.pkg
   25  -rw-    49898964   Aug 3 2013 15:05:40 +05:30
asr1000rp2-rpbase.03.12.01.S.154-2.S.pkg
   26  -rw-    46557940   Aug 3 2013 15:05:40 +05:30
asr1000rp2-rpcontrol.03.12.01.S.154-2.S.pkg
   27  -rw-   114612988   Aug 3 2013 15:05:41 +05:30
asr1000rp2-rpios-adventerprisek9.03.12.01.S.154-2.S.pkg
   28  -rw-    41954036   Aug 3 2013 15:05:41 +05:30
asr1000rp2-sipbase.03.12.01.S.154-2.S.pkg
   29  -rw-    60957428   Aug 3 2013 15:05:41 +05:30
asr1000rp2-sipspa.03.12.01.S.154-2.S.pkg
   19  -rw-        9838   Aug 3 2013 15:05:41 +05:30  packages.conf

1940303872 bytes total (503164928 bytes free)

Router# show redundancy states
       my state = 13 -ACTIVE
     peer state = 8  -STANDBY HOT
           Mode = Duplex
           Unit = Primary
        Unit ID = 48

Redundancy Mode (Operational) = sso
Redundancy Mode (Configured)  = sso
Redundancy State              = sso
     Maintenance Mode = Disabled
    Manual Swact = enabled
```

```
 Communications = Up

   client count = 108
 client_notification_TMR = 30000 milliseconds
           RF debug mask = 0x0

Router# copy running-config startup-config
Destination filename [startup-config]?
Building configuration...
[OK]
Router# mkdir harddisk:Target_Subs
Create directory filename [Target_Subs]?
Created dir harddisk:/Target_Subs
Router#
Router#
Router#
Router# request platform software package expand file
harddisk:Target_Subs/asr1000rp2-adventerprisek9.03.13.00.S.154-3.S-ext.bin to
harddisk:Target_Subs
Verifying parameters
Validating package type
Copying package files
SUCCESS: Finished expanding all-in-one software package.
Router#
Router#
Router# dir harddisk:Target_Subs
Directory of harddisk:/Target_Subs/

3358722  -rw-   569597380   Aug 4 2013 18:45:38 +05:30
asr1000rp2-adventerprisek9.03.13.00.S.154-3.S-ext.bin
7684099  -rw-    37557200   Aug 4 2013 18:46:43 +05:30
asr1000rp2-elcbase.03.13.00.S.154-3.S-ext.pkg
7684100  -rw-    51194832   Aug 4 2013 18:46:43 +05:30
asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg
7684101  -rw-    80657364   Aug 4 2013 18:46:43 +05:30
asr1000rp2-espbase.03.13.00.S.154-3.S-ext.pkg
7684102  -rw-    95446456   Aug 4 2013 18:46:43 +05:30
asr1000rp2-espx86base.03.13.00.S.154-3.S-ext.pkg
7684097  -rw-        9381   Aug 4 2013 18:46:43 +05:30
asr1000rp2-packages-adventerprisek9.03.13.00.S.154-3.S-ext.conf
7684103  -rw-    23350232   Aug 4 2013 18:46:43 +05:30
asr1000rp2-rpaccess.03.13.00.S.154-3.S-ext.pkg
7684104  -rw-    37694900   Aug 4 2013 18:46:44 +05:30
asr1000rp2-rpbase.03.13.00.S.154-3.S-ext.pkg
7684105  -rw-    45536216   Aug 4 2013 18:46:44 +05:30
asr1000rp2-rpcontrol.03.13.00.S.154-3.S-ext.pkg
7684106  -rw-   118754284   Aug 4 2013 18:46:44 +05:30
asr1000rp2-rpios-adventerprisek9.03.13.00.S.154-3.S-ext.pkg
7684107  -rw-    38380500   Aug 4 2013 18:46:44 +05:30
asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg
7684108  -rw-    61760468   Aug 4 2013 18:46:44 +05:30
asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg
7684098  -rw-       10165   Aug 4 2013 18:46:44 +05:30  packages.conf

78704144384 bytes total (9254879232 bytes free)
Router# copy harddisk:Target_Subs/asr1000rp2-espbase.03.13.00.S.154-3.S-ext.pkg
bootflash:Active_Dir/
Destination filename [Active_Dir/asr1000rp2-espbase.03.13.00.S.154-3.S-ext.pkg]?
Copy in
progress...CCCC
```

```
CCCC
```

```
Copy in
progress...CCCC
```

```
CCCC
```

```
CCCC
```

```
  Found asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg
Verifying image file locations
Inspecting image file types
    WARNING: In-service installation of IOSD package
    WARNING: requires software redundancy on target RP
    WARNING: or on-reboot parameter
    WARNING: Automatically setting the on-reboot flag
    WARNING: In-service installation of RP Base package
    WARNING: requires software reboot of target RP
Processing image file constraints
Creating candidate provisioning file
Finished image file verification

--- Starting candidate package set construction ---
Verifying existing software set
Processing candidate provisioning file
Constructing working set for candidate package set
Constructing working set for running package set
Checking command output
Constructing merge of running and candidate packages
Checking if resulting candidate package set would be complete
Finished candidate package set construction

--- Starting compatibility testing ---
Determining whether candidate package set is compatible
Determining whether installation is valid
Determining whether installation is valid ... skipped
Verifying image type compatibility
Checking IPC compatibility for candidate software
Checking candidate package set infrastructure compatibility
Checking infrastructure compatibility with running software
Checking infrastructure compatibility with running software ... skipped
Checking package specific compatibility
Finished compatibility testing

--- Starting mdr compatibility verification ---
Finished mdr compatibility verification

SUCCESS: Software is ISSU MDR compatible.
Router#
Router#
Router# issu loadversion rp 1 file
stby-bootflash:Active_Dir/asr1000rp*03.13.00.S.154-3.S-ext*.pkg force
--- Starting local lock acquisition on R0 ---
Finished local lock acquisition on R0

--- Starting installation state synchronization ---
Finished installation state synchronization

--- Starting local lock acquisition on R1 ---
Finished local lock acquisition on R1

--- Starting file path checking ---
Finished file path checking

--- Starting image file verification ---
Checking image file names
Locating image files and validating name syntax
  Found asr1000rp2-elcbase.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-espbase.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-espx86base.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-rpaccess.03.13.00.S.154-3.S-ext.pkg
```

```
  Found asr1000rp2-rpbase.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-rpcontrol.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-rpios-adventerprisek9.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg
Verifying image file locations
Inspecting image file types
    WARNING: In-service installation of IOSD package
    WARNING: requires software redundancy on target RP
    WARNING: or on-reboot parameter
    WARNING: Automatically setting the on-reboot flag
    WARNING: In-service installation of RP Base package
    WARNING: requires software reboot of target RP
Processing image file constraints
Creating candidate provisioning file
Finished image file verification

--- Starting candidate package set construction ---
Verifying existing software set
Processing candidate provisioning file
Constructing working set for candidate package set
Constructing working set for running package set
Checking command output
Constructing merge of running and candidate packages
Checking if resulting candidate package set would be complete
Finished candidate package set construction

--- Starting compatibility testing ---
Determining whether candidate package set is compatible
Determining whether installation is valid
Determining whether installation is valid ... skipped
Verifying image type compatibility
Checking IPC compatibility for candidate software
Checking candidate package set infrastructure compatibility
Checking infrastructure compatibility with running software
Checking infrastructure compatibility with running software ... skipped
Checking package specific compatibility
Finished compatibility testing

--- Starting list of software package changes ---
Old files list:
  Removed asr1000rp2-elcbase.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-elcspa.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-espbase.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-espx86base.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-rpaccess.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-rpbase.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-rpcontrol.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-rpios-adventerprisek9.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-sipbase.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-sipspa.03.12.01.S.154-2.S.pkg
New files list:
  Added asr1000rp2-elcbase.03.13.00.S.154-3.S-ext.pkg
  Added asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg
  Added asr1000rp2-espbase.03.13.00.S.154-3.S-ext.pkg
  Added asr1000rp2-espx86base.03.13.00.S.154-3.S-ext.pkg
  Added asr1000rp2-rpaccess.03.13.00.S.154-3.S-ext.pkg
  Added asr1000rp2-rpbase.03.13.00.S.154-3.S-ext.pkg
  Added asr1000rp2-rpcontrol.03.13.00.S.154-3.S-ext.pkg
  Added asr1000rp2-rpios-adventerprisek9.03.13.00.S.154-3.S-ext.pkg
  Added asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg
  Added asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg
Finished list of software package changes
```

```
--- Starting commit of software changes ---
Updating provisioning rollback files
Creating pending provisioning file
Committing provisioning file
Finished commit of software changes

SUCCESS: Software provisioned.  New software will load on reboot.
Router#
Router#
Router# hw-module slot r1 reload
Proceed with reload of module? [confirm]Y
Router#
*Aug  4 19:14:01.721 IST: %IOSXE_OIR-6-OFFLINECARD: Card (rp) offline in slot R1
*Aug  4 19:14:01.761 IST: %REDUNDANCY-3-STANDBY_LOST: Standby processor fault
(PEER_NOT_PRESENT)
*Aug  4 19:14:01.761 IST: %REDUNDANCY-3-STANDBY_LOST: Standby processor fault (PEER_DOWN)
*Aug  4 19:14:01.761 IST: %REDUNDANCY-3-STANDBY_LOST: Standby processor fault
(PEER_REDUNDANCY_STATE_CHANGE)
*Aug  4 19:14:03.584 IST: %RF-5-RF_RELOAD: Peer reload. Reason: EHSA standby down
*Aug  4 19:14:03.594 IST: % Redundancy mode change to SSO

Router#
*Aug  4 19:17:35.443 IST: %IOSXE_OIR-6-ONLINECARD: Card (rp) online in slot R1
Router#
*Aug  4 19:17:48.061 IST: %REDUNDANCY-5-PEER_MONITOR_EVENT: Active detected a standby
insertion (raw-event=PEER_FOUND(4))

*Aug  4 19:17:48.061 IST: %REDUNDANCY-5-PEER_MONITOR_EVENT: Active detected a standby
insertion (raw-event=PEER_REDUNDANCY_STATE_CHANGE(5))

*Aug  4 19:17:51.003 IST: %REDUNDANCY-3-IPC: IOS versions do not match.

*Aug  4 19:19:08.380 IST: %NBAR_HA-5-NBAR_INFO: NBAR sync DONE!
*Aug  4 19:19:08.797 IST: %HA_CONFIG_SYNC-6-BULK_CFGSYNC_SUCCEED: Bulk Sync succeeded
*Aug  4 19:19:08.798 IST: %RF-5-RF_TERMINAL_STATE: Terminal state reached for (SSO)
Router#
Router# issu loadversion rp 0 file
bootflash:Active_Dir/asr1000rp2-{sipbase,sipspa}*03.13.00.S.154-3.S-ext*.pkg slot 2 mdr
force
--- Starting local lock acquisition on R0 ---
Finished local lock acquisition on R0

--- Starting installation state synchronization ---
Finished installation state synchronization

--- Starting file path checking ---
Finished file path checking

--- Starting image file verification ---
Checking image file names
Locating image files and validating name syntax
  Found asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg
Verifying image file locations
Inspecting image file types
Processing image file constraints
Creating candidate provisioning file
Finished image file verification

--- Starting candidate package set construction ---
Verifying existing software set
Processing candidate provisioning file
Constructing working set for candidate package set
Constructing working set for running package set
```

```
Checking command output
Constructing merge of running and candidate packages
Checking if resulting candidate package set would be complete
Finished candidate package set construction

--- Starting compatibility testing ---
Determining whether candidate package set is compatible

WARNING:
WARNING: Candidate software combination not found in compatibility database
WARNING:

Determining whether installation is valid
Creating matrix_file by locate_latest_matrix_file  /tmp/issu/provision/sw

WARNING:
WARNING: Candidate software combination not found in compatibility database
WARNING:


WARNING:
WARNING: Candidate software combination not found in compatibility database
WARNING:

Software sets are identified as compatible
Verifying image type compatibility
Checking IPC compatibility with running software
Checking candidate package set infrastructure compatibility
Checking infrastructure compatibility with running software
Checking package specific compatibility
Finished compatibility testing

--- Starting mdr compatibility verification ---
Finished mdr compatibility verification

--- Starting impact testing ---
Checking operational impact of change
Finished impact testing

--- Starting list of software package changes ---
No old package files removed
New files list:
  Added asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg
  Added asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg
Finished list of software package changes

--- Starting commit of software changes ---
Updating provisioning rollback files
Creating pending provisioning file
Committing provisioning file
Finished commit of software changes

--- Starting analysis of software changes ---
Finished analysis of software changes

--- Starting update running software ---
Blocking peer synchronization of operating information
Creating the command set placeholder directory
  Finding latest command set
  Finding latest command shortlist lookup file
  Finding latest command shortlist file
  Assembling CLI output libraries
  Assembling CLI input libraries
  Assembling Dynamic configuration files
```

```
  Applying interim IPC and database definitions
  Replacing running software
  Replacing CLI software
  Restarting software

Aug  4 19:20:48.949 IST: %MDR-5-CARD_RESTART: R0/0: card_mdr:  Minimal Disruptive Restart
SIP2 to acquire provisioned software  Applying final IPC and database definitions

*Aug  4 19:20:58.013 IST: %CMCC-5-SPA_MDR_INIT: SIP2: cmcc:  SPA0 initiated Minimal
Disruptive Restart
*Aug  4 19:20:58.017 IST: %CMCC-5-SPA_MDR_INIT: SIP2: cmcc:  SPA1 initiated Minimal
Disruptive Restart
*Aug  4 19:20:58.018 IST: %CMCC-5-SPA_MDR_INIT: SIP2: cmcc:  SPA2 initiated Minimal
Disruptive Restart
*Aug  4 19:20:58.019 IST: %CMCC-5-SPA_MDR_INIT: SIP2: cmcc:  SPA3 initiated Minimal
Disruptive Restart  Generating software version information
  Notifying running software of updates
  Unblocking peer synchronization of operating information
Unmounting old packages
Cleaning temporary installation files
  Finished update running software

SUCCESS: Finished installing software.

Router#
*Aug  4 19:21:45.424 IST: %IOSXE_OIR-6-ONLINECARD: Card (cc) online in slot 2
*Aug  4 19:21:48.382 IST: %IOSXE_OIR-6-INSSPA: SPA inserted in subslot 2/0
*Aug  4 19:21:48.733 IST: %IOSXE_OIR-6-INSSPA: SPA inserted in subslot 2/1
*Aug  4 19:21:49.083 IST: %IOSXE_OIR-6-INSSPA: SPA inserted in subslot 2/2
*Aug  4 19:21:49.430 IST: %IOSXE_OIR-6-INSSPA: SPA inserted in subslot 2/3
*Aug  4 19:21:58.121 IST: %LINK-3-UPDOWN: SIP2/0: Interface EOBC2/1, changed state to up
*Aug  4 19:22:02.302 IST: %SPA_OIR-6-ONLINECARD: SPA (SPA-1X10GE-L-V2) online in subslot
2/0
*Aug  4 19:22:02.312 IST: %CMCC-5-SPA_MDR_DONE: SIP2: cmcc:  SPA0 completed Minimal
Disruptive Restart
*Aug  4 19:22:02.282 IST: %TRANSCEIVER-6-INSERTED: SIP2/0: transceiver module inserted in
TenGigabitEthernet2/0/0
*Aug  4 19:22:02.518 IST: %LINK-3-UPDOWN: SIP2/1: Interface EOBC2/1, changed state to up
*Aug  4 19:22:06.113 IST: %SPA_OIR-6-ONLINECARD: SPA (SPA-1X10GE-L-V2) online in subslot
2/1
*Aug  4 19:22:06.126 IST: %CMCC-5-SPA_MDR_DONE: SIP2: cmcc:  SPA1 completed Minimal
Disruptive Restart
*Aug  4 19:22:06.082 IST: %TRANSCEIVER-6-INSERTED: SIP2/1: transceiver module inserted in
TenGigabitEthernet2/1/0
*Aug  4 19:22:08.080 IST: %LINK-3-UPDOWN: SIP2/2: Interface EOBC2/1, changed state to up
*Aug  4 19:22:11.627 IST: %SPA_OIR-6-ONLINECARD: SPA (SPA-1X10GE-L-V2) online in subslot
2/2
*Aug  4 19:22:11.637 IST: %CMCC-5-SPA_MDR_DONE: SIP2: cmcc:  SPA2 completed Minimal
Disruptive Restart
*Aug  4 19:22:11.616 IST: %TRANSCEIVER-6-INSERTED: SIP2/2: transceiver module inserted in
TenGigabitEthernet2/2/0
*Aug  4 19:22:12.523 IST: %LINK-3-UPDOWN: SIP2/3: Interface EOBC2/1, changed state to up
*Aug  4 19:22:16.657 IST: %SPA_OIR-6-ONLINECARD: SPA (SPA-1X10GE-L-V2) online in subslot
2/3
*Aug  4 19:22:16.668 IST: %CMCC-5-SPA_MDR_DONE: SIP2: cmcc:  SPA3 completed Minimal
Disruptive Restart
*Aug  4 19:22:16.648 IST: %TRANSCEIVER-6-INSERTED: SIP2/3: transceiver module inserted in
TenGigabitEthernet2/3/0
Router#
Router#
Router# issu commitversion
--- Starting local lock acquisition on R0 ---
Finished local lock acquisition on R0
```

```
--- Starting installation changes ---
Cancelling rollback timer
Finished installation changes

SUCCESS: Installation changes committed
Router#
Router#
Router#
Router#issu loadversion rp 0 file
harddisk:boot_dir/asr1000rp2-{elcbase,elcspa}*03.13.00.S.154-3.S.pkg slot 4 mdr
--- Starting local lock acquisition on R0 ---
Finished local lock acquisition on R0

--- Starting installation state synchronization ---
Finished installation state synchronization

--- Starting file path checking ---
Finished file path checking

--- Starting image file verification ---
Checking image file names
Locating image files and validating name syntax
  Found asr1000rp2-elcbase.03.13.00.S.154-3.pkg
  Found asr1000rp2-elcspa.03.13.00.S.154-3.pkg
Verifying image file locations
Inspecting image file types
Processing image file constraints
Creating candidate provisioning file
Finished image file verification

--- Starting candidate package set construction ---
Verifying existing software set
Processing candidate provisioning file
Constructing working set for candidate package set
Constructing working set for running package set
Checking command output
Constructing merge of running and candidate packages
Checking if resulting candidate package set would be complete
Finished candidate package set construction

--- Starting compatibility testing ---
Determining whether candidate package set is compatible

WARNING:
WARNING: Candidate software combination not found in compatibility database
WARNING:

Determining whether installation is valid
Creating matrix_file by locate_latest_matrix_file  /tmp/issu/provision/sw

WARNING:
WARNING: Candidate software combination not found in compatibility database
WARNING:


WARNING:
WARNING: Candidate software combination not found in compatibility database
WARNING:

Software sets are identified as compatible
Verifying image type compatibility
Checking IPC compatibility with running software
Checking candidate package set infrastructure compatibility
Checking infrastructure compatibility with running software
```

```
Checking package specific compatibility
Finished compatibility testing

--- Starting mdr compatibility verification ---

WARNING:
WARNING: ISSU between engineering builds with release strings in non-standard format.
Skipping MDR Software Compatibility checks.
WARNING:


WARNING:
WARNING: ISSU between engineering builds with release strings in non-standard format.
Skipping MDR Software Compatibility checks.
WARNING:

Finished mdr compatibility verification

--- Starting impact testing ---
Checking operational impact of change
Finished impact testing

--- Starting list of software package changes ---
No old package files removed
New files list:
  Added asr1000rp2-elcbase.03.13.00.S.154-3.pkg
  Added asr1000rp2-elcspa.03.13.00.S.154-3.pkg
Finished list of software package changes

--- Starting commit of software changes ---
Updating provisioning rollback files
Creating pending provisioning file
Committing provisioning file
Finished commit of software changes

--- Starting analysis of software changes ---
Finished analysis of software changes

--- Starting update running software ---
Blocking peer synchronization of operating information
Creating the command set placeholder directory
  Finding latest command set
  Finding latest command shortlist lookup file
  Finding latest command shortlist file
  Assembling CLI output libraries
  Assembling CLI input libraries
  Assembling Dynamic configuration files
  Applying interim IPC and database definitions
  Replacing running software
  Replacing CLI software
  Restarting software
  Applying final IPC and database definitions

*Jan 13 00:41:37.778 PST: %MDR-5-CARD_RESTART: R0/0: card_mdr:  Minimal Disruptive Restart
SIP0 to acquire provisioned software
*Jan 13 00:41:47.894 PST: %CMCC-5-SPA_MDR_INIT: SIP0: cmcc:  SPA0 initiated Minimal
Disruptive Restart  Generating softwareversion information
  Notifying running software of updates
  Unblocking peer synchronization of operating information
Unmounting old packages
Cleaning temporary installation files
  Finished update running software

SUCCESS: Finished installing software.
```

```
Router#
Router#
*Jan 13 00:43:15.162 PST: %IOSXE_OIR-6-ONLINECARD: Card (cc) online in slot 4
*Jan 13 00:43:18.773 PST: %IOSXE_OIR-6-INSSPA: SPA inserted in subslot 4/0
*Jan 13 00:43:37.922 PST: %LINK-3-UPDOWN: SIP4/0: Interface EOBC0/1, changed state to up
*Jan 13 00:43:46.366 PST: %SPA_OIR-6-ONLINECARD: SPA (BUILT-IN-2T+20X1GE) online in
subslot 4/0
*Jan 13 00:43:45.525 PST: %CMCC-5-SPA_MDR_DONE: SIP4: cmcc: SPA0 completed Minimal
Disruptive Restart
Router#
Router#
Router#
Router#issu commitversion
--- Starting local lock acquisition on R0 ---
Finished local lock acquisition on R0

--- Starting installation changes ---
Cancelling rollback timer
Finished installation changes

SUCCESS: Installation changes committed
Router#
Router#
Router# issu loadversion rp 0 file
bootflash:Active_Dir/asr1000rp2-esp*03.13.00.S.154-3.S-ext*.pkg slot 1
--- Starting local lock acquisition on R0 ---
Finished local lock acquisition on R0

--- Starting installation state synchronization ---
Finished installation state synchronization

--- Starting file path checking ---
Finished file path checking

--- Starting image file verification ---
Checking image file names
Locating image files and validating name syntax
  Found asr1000rp2-espbase.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-espx86base.03.13.00.S.154-3.S-ext.pkg
Verifying image file locations
Inspecting image file types
Processing image file constraints
Creating candidate provisioning file
Finished image file verification

--- Starting candidate package set construction ---
Verifying existing software set
Processing candidate provisioning file
Constructing working set for candidate package set
Constructing working set for running package set
Checking command output
Constructing merge of running and candidate packages
Checking if resulting candidate package set would be complete
Finished candidate package set construction

--- Starting compatibility testing ---
Determining whether candidate package set is compatible

WARNING:
WARNING: Candidate software combination not found in compatibility database
WARNING:
```

```
Determining whether installation is valid
Creating matrix_file by locate_latest_matrix_file  /tmp/issu/provision/sw

WARNING:
WARNING: Candidate software combination not found in compatibility database
WARNING:


WARNING:
WARNING: Candidate software combination not found in compatibility database
WARNING:

Software sets are identified as compatible
Verifying image type compatibility
Checking IPC compatibility with running software
Checking candidate package set infrastructure compatibility
Checking infrastructure compatibility with running software
Checking package specific compatibility
Finished compatibility testing

--- Starting impact testing ---
Checking operational impact of change
Finished impact testing

--- Starting list of software package changes ---
No old package files removed
New files list:
  Added asr1000rp2-espbase.03.13.00.S.154-3.S-ext.pkg
  Added asr1000rp2-espx86base.03.13.00.S.154-3.S-ext.pkg
Finished list of software package changes

--- Starting commit of software changes ---
Updating provisioning rollback files
Creating pending provisioning file
Committing provisioning file
Finished commit of software changes

--- Starting analysis of software changes ---
Finished analysis of software changes

--- Starting update running software ---
Blocking peer synchronization of operating information
Creating the command set placeholder directory
  Finding latest command set
  Finding latest command shortlist lookup file
  Finding latest command shortlist file
  Assembling CLI output libraries
  Assembling CLI input libraries
  Assembling Dynamic configuration files
  Applying interim IPC and database definitions
  Replacing running software
  Replacing CLI software
  Restarting software
    Restarting ESP1
  Applying final IPC and database definitions

Aug  4 19:29:16.751 IST: %IOSXE_OIR-6-OFFLINECARD: Card (fp) offline in slot F1
*Aug  4 19:29:18.172 IST: %CMRP-6-FP_HA_STATUS: R0/0: cmand:  F0 redundancy state is
Active with no Standby  Generating software version information
  Notifying running software of updates
  Unblocking peer synchronization of operating information
Unmounting old packages
Cleaning temporary installation files
  Finished update running software
```

```
SUCCESS: Finished installing software.

Router#
*Aug  4 19:30:50.972 IST: %CPPHA-7-START: F1: cpp_ha:  CPP 0 preparing image
/tmp/sw/fp/1/0/fpx86/mount/usr/cpp/bin/qfp-ucode-esp40
*Aug  4 19:30:51.362 IST: %CPPHA-7-START: F1: cpp_ha:  CPP 0 startup init image
/tmp/sw/fp/1/0/fpx86/mount/usr/cpp/bin/qfp-ucode-esp40
*Aug  4 19:30:53.088 IST: %IOSXE_OIR-6-ONLINECARD: Card (fp) online in slot F1
Router#
Router# issu commitversion
--- Starting local lock acquisition on R0 ---
Finished local lock acquisition on R0

--- Starting installation changes ---
Cancelling rollback timer
Finished installation changes

SUCCESS: Installation changes committed
Router#
Router#
Router# issu loadversion rp 0 file
bootflash:Active_Dir/asr1000rp2-esp*03.13.00.S.154-3.S-ext*.pkg slot 0
--- Starting local lock acquisition on R0 ---
Finished local lock acquisition on R0

--- Starting installation state synchronization ---
Finished installation state synchronization

--- Starting file path checking ---
Finished file path checking

--- Starting image file verification ---
Checking image file names
Locating image files and validating name syntax
  Found asr1000rp2-espbase.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-espx86base.03.13.00.S.154-3.S-ext.pkg
Verifying image file locations
Inspecting image file types
Processing image file constraints
Creating candidate provisioning file

Aug  4 19:31:14.730 IST: %CPPHA-7-START: F1: cpp_ha:  CPP 0 running init image
/tmp/sw/fp/1/0/fpx86/mount/usr/cpp/bin/qfp-ucode-esp40
*Aug  4 19:31:15.079 IST: %CPPHA-7-READY: F1: cpp_ha:  CPP 0 loading and initialization
completeFinished image file verification

--- Starting candidate package set construction ---
Verifying existing software set

*Aug  4 19:31:15.309 IST: %IOSXE-6-PLATFORM: F1: cpp_cp: Process
CPP_PFILTER_EA_EVENT__API_CALL__REGISTERProcessing candidate provisioning file

*Aug  4 19:31:18.010 IST: %CMRP-6-FP_HA_STATUS: R0/0: cmand:  F0 redundancy state is
Active with ready StandbyConstructing working set for candidate package set
Constructing working set for running package set
Checking command output
Constructing merge of running and candidate packages
Checking if resulting candidate package set would be complete
Finished candidate package set construction

--- Starting compatibility testing ---
Determining whether candidate package set is compatible
```

```
WARNING:
WARNING: Candidate software combination not found in compatibility database
WARNING:

Determining whether installation is valid
Creating matrix_file by locate_latest_matrix_file  /tmp/issu/provision/sw
Software sets are identified as compatible
Verifying image type compatibility
Checking IPC compatibility with running software
Checking candidate package set infrastructure compatibility
Checking infrastructure compatibility with running software
Checking package specific compatibility
Finished compatibility testing

--- Starting impact testing ---
Checking operational impact of change
Finished impact testing

--- Starting list of software package changes ---
Old files list:
  Removed asr1000rp2-espbase.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-espx86base.03.12.01.S.154-2.S.pkg
No new package files added
Finished list of software package changes

--- Starting commit of software changes ---
Updating provisioning rollback files
Creating pending provisioning file
Committing provisioning file
Finished commit of software changes

--- Starting analysis of software changes ---
Finished analysis of software changes

--- Starting update running software ---
Blocking peer synchronization of operating information
Creating the command set placeholder directory
  Finding latest command set
  Finding latest command shortlist lookup file
  Finding latest command shortlist file
  Assembling CLI output libraries
  Assembling CLI input libraries
  Assembling Dynamic configuration files
  Applying interim IPC and database definitions
  Replacing running software
  Replacing CLI software
  Restarting software
    Restarting ESP0
  Applying final IPC and database definitions

*Aug  4 19:32:46.187 IST: %IOSXE_OIR-6-OFFLINECARD: Card (fp) offline in slot F0
*Aug  4 19:32:46.539 IST: %CMRP-6-FP_HA_STATUS: R0/0: cmand:  F1 redundancy state is
Active  Generating software version information
  Notifying running software of updates
  Unblocking peer synchronization of operating information
Unmounting old packages
Cleaning temporary installation files
  Finished update running software

SUCCESS: Finished installing software.

Router#
*Aug  4 19:34:19.748 IST: %CPPHA-7-START: F0: cpp_ha:  CPP 0 preparing image
/tmp/sw/fp/0/0/fpx86/mount/usr/cpp/bin/qfp-ucode-esp40
```

```
*Aug  4 19:34:20.139 IST: %CPPHA-7-START: F0: cpp_ha:  CPP 0 startup init image
/tmp/sw/fp/0/0/fpx86/mount/usr/cpp/bin/qfp-ucode-esp40
*Aug  4 19:34:21.858 IST: %IOSXE_OIR-6-ONLINECARD: Card (fp) online in slot F0
*Aug  4 19:34:43.609 IST: %CPPHA-7-START: F0: cpp_ha:  CPP 0 running init image
/tmp/sw/fp/0/0/fpx86/mount/usr/cpp/bin/qfp-ucode-esp40
*Aug  4 19:34:43.958 IST: %CPPHA-7-READY: F0: cpp_ha:  CPP 0 loading and initialization
complete
*Aug  4 19:34:44.190 IST: %IOSXE-6-PLATFORM: F0: cpp_cp: Process
CPP_PFILTER_EA_EVENT__API_CALL__REGISTER
*Aug  4 19:34:46.890 IST: %CMRP-6-FP_HA_STATUS: R0/0: cmand:  F0 redundancy state is
Standby
Router#
Router# show platform
Chassis type: ASR1013

Slot      Type                State                   Insert time (ago)
--------- ------------------- ---------------------- -----------------
 2        ASR1000-SIP40        ok                     1d04h
 2/0      SPA-1X10GE-L-V2      ok                     1d04h
 2/1      SPA-1X10GE-L-V2      ok                     1d04h
 2/2      SPA-1X10GE-L-V2      ok                     1d04h
 2/3       SPA-1X10GE-L-V2      ok                      1d04h
 4         ASR1000-2T+20X1GE    ok                      1d04h
 4/0       BUILT-IN-2T+20X1GE   ok                      1d04h
 R0        ASR1000-RP2          ok, active              1d04h
 R1        ASR1000-RP2          ok, standby             1d04h
 F0        ASR1000-ESP100       ok, standby             1d04h
 F1        ASR1000-ESP100       ok, active              1d04h
 P0        ASR1013-PWR-AC       ok                      1d04h
 P1        ASR1013-PWR-AC       ok                      1d04h
 P2        ASR1013-PWR-AC       ok                      1d04h
 P3        ASR1013-PWR-AC       ps, fail                1d04h

Slot      CPLD Version        Firmware Version
--------- ------------------- ----------------------------------------
2         00200800            15.3(3r)S
4         00200800            15.3(1r)S
R0        10021901            15.3(3r)S
R1        10021901            15.3(3r)S
F0        12071700            15.3(3r)S
F1        12071700            15.3(3r)S

Router#
Router# issu commitversion
--- Starting local lock acquisition on R0 ---
Finished local lock acquisition on R0

--- Starting installation changes ---
Cancelling rollback timer
Finished installation changes

SUCCESS: Installation changes committed
Router#
Router#
Router#issu loadversion rp 0 file
bootflash:Active_Dir/asr1000rp2*03.13.00.S.154-3.S-ext*.pkg force
--- Starting local lock acquisition on R0 ---
Finished local lock acquisition on R0

--- Starting installation state synchronization ---
Finished installation state synchronization

--- Starting file path checking ---
Finished file path checking
```

```
--- Starting image file verification ---
Checking image file names
Locating image files and validating name syntax
  Found asr1000rp2-elcbase.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-espbase.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-espx86base.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-rpaccess.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-rpbase.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-rpcontrol.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-rpios-adventerprisek9.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg
Verifying image file locations
Inspecting image file types
    WARNING: In-service installation of IOSD package
    WARNING: requires software redundancy on target RP
    WARNING: or on-reboot parameter
    WARNING: Automatically setting the on-reboot flag
    WARNING: In-service installation of RP Base package
    WARNING: requires software reboot of target RP
Processing image file constraints
Creating candidate provisioning file
Finished image file verification

--- Starting candidate package set construction ---
Verifying existing software set
Processing candidate provisioning file
Constructing working set for candidate package set
Constructing working set for running package set
Checking command output
Constructing merge of running and candidate packages
Checking if resulting candidate package set would be complete
Finished candidate package set construction

--- Starting compatibility testing ---
Determining whether candidate package set is compatible
Determining whether installation is valid
Determining whether installation is valid ... skipped
Verifying image type compatibility
Checking IPC compatibility for candidate software
Checking candidate package set infrastructure compatibility
Checking infrastructure compatibility with running software
Checking infrastructure compatibility with running software ... skipped
Checking package specific compatibility
Finished compatibility testing

--- Starting list of software package changes ---
Old files list:
  Removed asr1000rp2-elcbase.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-elcspa.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-rpaccess.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-rpbase.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-rpcontrol.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-rpios-adventerprisek9.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-sipbase.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-sipspa.03.12.01.S.154-2.S.pkg
New files list:
  Added asr1000rp2-rpaccess.03.13.00.S.154-3.S-ext.pkg
  Added asr1000rp2-rpbase.03.13.00.S.154-3.S-ext.pkg
  Added asr1000rp2-rpcontrol.03.13.00.S.154-3.S-ext.pkg
  Added asr1000rp2-rpios-adventerprisek9.03.13.00.S.154-3.S-ext.pkg
Finished list of software package changes
```

```
--- Starting commit of software changes ---
Updating provisioning rollback files
Creating pending provisioning file
Committing provisioning file
Finished commit of software changes

SUCCESS: Software provisioned.  New software will load on reboot.
Router#
Router#
Router# show version R0 provisioned
Package: Provisioning File, version: n/a, status: active
  File: bootflash:Active_Dir/packages.conf, on: RP0
  Built: n/a, by: n/a
  File SHA1 checksum: c79075780592aec1312725f4a2357a034fda2d3b

Package: rpbase, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-rpbase.03.13.00.S.154-3.S-ext.pkg, on: RP0
  Built: 2013-07-25_22.55, by: mcpre
  File SHA1 checksum: 4f655c54bb95b4dfa24a0d25ebf97cf8527c69e9

Package: rpcontrol, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-rpcontrol.03.13.00.S.154-3.S-ext.pkg, on: RP0/0
  Built: 2013-07-25_22.55, by: mcpre
  File SHA1 checksum: 8a0a45ea5c7a656c0eef6726174461584f182c78

Package: rpios-adventerprisek9, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-rpios-adventerprisek9.03.13.00.S.154-3.S-ext.pkg,
on: RP0/0
  Built: 2013-07-25_23.00, by: mcpre
  File SHA1 checksum: 85e9eab826bff2194ef568a56c76453625383ad2

Package: rpaccess, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-rpaccess.03.13.00.S.154-3.S-ext.pkg, on: RP0/0
  Built: 2013-07-25_22.55, by: mcpre
  File SHA1 checksum: a360dff0fd76a9b1ae67cda9116c97b62f25ab09

Package: rpcontrol, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-rpcontrol.03.13.00.S.154-3.S-ext.pkg, on: RP0/1
  Built: 2013-07-25_22.55, by: mcpre
  File SHA1 checksum: 8a0a45ea5c7a656c0eef6726174461584f182c78

Package: rpios-adventerprisek9, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-rpios-adventerprisek9.03.13.00.S.154-3.S-ext.pkg,
on: RP0/1
  Built: 2013-07-25_23.00, by: mcpre
  File SHA1 checksum: 85e9eab826bff2194ef568a56c76453625383ad2

Package: rpaccess, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-rpaccess.03.13.00.S.154-3.S-ext.pkg, on: RP0/1
  Built: 2013-07-25_22.55, by: mcpre
  File SHA1 checksum: a360dff0fd76a9b1ae67cda9116c97b62f25ab09

Package: rpbase, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-rpbase.03.13.00.S.154-3.S-ext.pkg, on: RP1
  Built: 2013-07-25_22.55, by: mcpre
  File SHA1 checksum: 4f655c54bb95b4dfa24a0d25ebf97cf8527c69e9

Package: rpcontrol, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-rpcontrol.03.13.00.S.154-3.S-ext.pkg, on: RP1/0
  Built: 2013-07-25_22.55, by: mcpre
  File SHA1 checksum: 8a0a45ea5c7a656c0eef6726174461584f182c78

Package: rpios-adventerprisek9, version: 03.13.00.S.154-3.S-ext, status: n/a
```

```
  File: bootflash:Active_Dir/asr1000rp2-rpios-adventerprisek9.03.13.00.S.154-3.S-ext.pkg,
on: RP1/0
  Built: 2013-07-25_23.00, by: mcpre
  File SHA1 checksum: 85e9eab826bff2194ef568a56c76453625383ad2

Package: rpaccess, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-rpaccess.03.13.00.S.154-3.S-ext.pkg, on: RP1/0
  Built: 2013-07-25_22.55, by: mcpre
  File SHA1 checksum: a360dff0fd76a9b1ae67cda9116c97b62f25ab09

Package: rpcontrol, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-rpcontrol.03.13.00.S.154-3.S-ext.pkg, on: RP1/1
  Built: 2013-07-25_22.55, by: mcpre
  File SHA1 checksum: 8a0a45ea5c7a656c0eef6726174461584f182c78

Package: rpios-adventerprisek9, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-rpios-adventerprisek9.03.13.00.S.154-3.S-ext.pkg,
on: RP1/1
  Built: 2013-07-25_23.00, by: mcpre
  File SHA1 checksum: 85e9eab826bff2194ef568a56c76453625383ad2

Package: rpaccess, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-rpaccess.03.13.00.S.154-3.S-ext.pkg, on: RP1/1
  Built: 2013-07-25_22.55, by: mcpre
  File SHA1 checksum: a360dff0fd76a9b1ae67cda9116c97b62f25ab09

Package: espbase, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-espbase.03.13.00.S.154-3.S-ext.pkg, on: ESP0
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 2fe0ede1545e3f8260b7d453653e812500f0d7b0

Package: espx86base, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-espx86base.03.13.00.S.154-3.S-ext.pkg, on: ESP0
  Built: 2013-07-25_22.55, by: mcpre
  File SHA1 checksum: 571b8bb3866341badd6e24de677b98409f0c789c

Package: espbase, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-espbase.03.13.00.S.154-3.S-ext.pkg, on: ESP1
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 2fe0ede1545e3f8260b7d453653e812500f0d7b0

Package: espx86base, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-espx86base.03.13.00.S.154-3.S-ext.pkg, on: ESP1
  Built: 2013-07-25_22.55, by: mcpre
  File SHA1 checksum: 571b8bb3866341badd6e24de677b98409f0c789c

Package: sipbase, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg, on: SIP0
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 3b6a4838972840a995ff22e73fd2bae910b268a7

Package: elcbase, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcbase.03.13.00.S.154-3.S-ext.pkg, on: SIP0
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 99f8dc925083b118626a4e82d93079050db96826

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP0/0
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP0/0
  Built: 2013-07-25_21.16, by: mcpre
```

```
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP0/1
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP0/1
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP0/2
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP0/2
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP0/3
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP0/3
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipbase, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg, on: SIP1
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 3b6a4838972840a995ff22e73fd2bae910b268a7

Package: elcbase, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcbase.03.13.00.S.154-3.S-ext.pkg, on: SIP1
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 99f8dc925083b118626a4e82d93079050db96826

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP1/0
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP1/0
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP1/1
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP1/1
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP1/2
```

```
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP1/2
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP1/3
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP1/3
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipbase, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg, on: SIP2
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 3b6a4838972840a995ff22e73fd2bae910b268a7

Package: elcbase, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcbase.03.13.00.S.154-3.S-ext.pkg, on: SIP2
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 99f8dc925083b118626a4e82d93079050db96826

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP2/0
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP2/0
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP2/1
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP2/1
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP2/2
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP2/2
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP2/3
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
```

```
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP2/3
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipbase, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg, on: SIP3
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 3b6a4838972840a995ff22e73fd2bae910b268a7

Package: elcbase, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcbase.03.13.00.S.154-3.S-ext.pkg, on: SIP3
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 99f8dc925083b118626a4e82d93079050db96826

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP3/0
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP3/0
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP3/1
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP3/1
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP3/2
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP3/2
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP3/3
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP3/3
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipbase, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg, on: SIP4
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 3b6a4838972840a995ff22e73fd2bae910b268a7

Package: elcbase, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcbase.03.13.00.S.154-3.S-ext.pkg, on: SIP4
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 99f8dc925083b118626a4e82d93079050db96826
```

```
Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP4/0
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP4/0
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP4/1
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP4/1
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP4/2
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP4/2
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP4/3
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP4/3
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipbase, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg, on: SIP5
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 3b6a4838972840a995ff22e73fd2bae910b268a7

Package: elcbase, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcbase.03.13.00.S.154-3.S-ext.pkg, on: SIP5
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 99f8dc925083b118626a4e82d93079050db96826

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP5/0
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP5/0
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP5/1
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca
```

```
Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP5/1
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP5/2
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP5/2
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP5/3
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP5/3
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499


Router#
Router#
Router# show version R0 provisioned
Package: Provisioning File, version: n/a, status: active
  File: bootflash:Active_Dir/packages.conf, on: RP0
  Built: n/a, by: n/a
  File SHA1 checksum: c79075780592aec1312725f4a2357a034fda2d3b

Package: rpbase, version: 03.12.01.S.154-2.S, status: active
  File: bootflash:Active_Dir/asr1000rp2-rpbase.03.12.01.S.154-2.S.pkg, on: RP0
  Built: 2013-03-25_18.48, by: mcpre
  File SHA1 checksum: 3a9675142898cfac350d4e42f0e37bd9f4e48538

Package: rpcontrol, version: 03.12.01.S.154-2.S, status: active
  File: bootflash:Active_Dir/asr1000rp2-rpcontrol.03.12.01.S.154-2.S.pkg, on: RP0/0
  Built: 2013-03-25_18.48, by: mcpre
  File SHA1 checksum: 87b11f863f67fdf2610ee0769b929baab4c3efad

Package: rpios-adventerprisek9, version: 03.12.01.S.154-2.S, status: active
  File: bootflash:Active_Dir/asr1000rp2-rpios-adventerprisek9.03.12.01.S.154-2.S.pkg, on:
RP0/0
  Built: 2013-03-25_18.51, by: mcpre
  File SHA1 checksum: b487136319da0a327844d353c77e533c53c56053

Package: rpaccess, version: 03.12.01.S.154-2.S, status: active
  File: bootflash:Active_Dir/asr1000rp2-rpaccess.03.12.01.S.154-2.S.pkg, on: RP0/0
  Built: 2013-03-25_18.48, by: mcpre
  File SHA1 checksum: 032bea36f74b19977b363243c99f02413b54104d

Package: rpcontrol, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-rpcontrol.03.12.01.S.154-2.S.pkg, on: RP0/1
  Built: 2013-03-25_18.48, by: mcpre
  File SHA1 checksum: 87b11f863f67fdf2610ee0769b929baab4c3efad

Package: rpios-adventerprisek9, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-rpios-adventerprisek9.03.12.01.S.154-2.S.pkg, on:
RP0/1
```

```
  Built: 2013-03-25_18.51, by: mcpre
  File SHA1 checksum: b487136319da0a327844d353c77e533c53c56053

Package: rpaccess, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-rpaccess.03.12.01.S.154-2.S.pkg, on: RP0/1
  Built: 2013-03-25_18.48, by: mcpre
  File SHA1 checksum: 032bea36f74b19977b363243c99f02413b54104d

Package: rpbase, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-rpbase.03.12.01.S.154-2.S.pkg, on: RP1
  Built: 2013-03-25_18.48, by: mcpre
  File SHA1 checksum: 3a9675142898cfac350d4e42f0e37bd9f4e48538

Package: rpcontrol, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-rpcontrol.03.12.01.S.154-2.S.pkg, on: RP1/0
  Built: 2013-03-25_18.48, by: mcpre
  File SHA1 checksum: 87b11f863f67fdf2610ee0769b929baab4c3efad

Package: rpios-adventerprisek9, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-rpios-adventerprisek9.03.12.01.S.154-2.S.pkg, on:
RP1/0
  Built: 2013-03-25_18.51, by: mcpre
  File SHA1 checksum: b487136319da0a327844d353c77e533c53c56053

Package: rpaccess, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-rpaccess.03.12.01.S.154-2.S.pkg, on: RP1/0
  Built: 2013-03-25_18.48, by: mcpre
  File SHA1 checksum: 032bea36f74b19977b363243c99f02413b54104d

Package: rpcontrol, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-rpcontrol.03.12.01.S.154-2.S.pkg, on: RP1/1
  Built: 2013-03-25_18.48, by: mcpre
  File SHA1 checksum: 87b11f863f67fdf2610ee0769b929baab4c3efad

Package: rpios-adventerprisek9, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-rpios-adventerprisek9.03.12.01.S.154-2.S.pkg, on:
RP1/1
  Built: 2013-03-25_18.51, by: mcpre
  File SHA1 checksum: b487136319da0a327844d353c77e533c53c56053

Package: rpaccess, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-rpaccess.03.12.01.S.154-2.S.pkg, on: RP1/1
  Built: 2013-03-25_18.48, by: mcpre
  File SHA1 checksum: 032bea36f74b19977b363243c99f02413b54104d

Package: espbase, version: 03.13.00.S.154-3.S-ext, status: inactive
  File: bootflash:Active_Dir/asr1000rp2-espbase.03.13.00.S.154-3.S-ext.pkg, on: ESP0
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 2fe0ede1545e3f8260b7d453653e812500f0d7b0

Package: espx86base, version: 03.13.00.S.154-3.S-ext, status: active
  File: bootflash:Active_Dir/asr1000rp2-espx86base.03.13.00.S.154-3.S-ext.pkg, on: ESP0
  Built: 2013-07-25_22.55, by: mcpre
  File SHA1 checksum: 571b8bb3866341badd6e24de677b98409f0c789c

Package: espbase, version: 03.13.00.S.154-3.S-ext, status: inactive
  File: bootflash:Active_Dir/asr1000rp2-espbase.03.13.00.S.154-3.S-ext.pkg, on: ESP1
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 2fe0ede1545e3f8260b7d453653e812500f0d7b0

Package: espx86base, version: 03.13.00.S.154-3.S-ext, status: active
  File: bootflash:Active_Dir/asr1000rp2-espx86base.03.13.00.S.154-3.S-ext.pkg, on: ESP1
  Built: 2013-07-25_22.55, by: mcpre
  File SHA1 checksum: 571b8bb3866341badd6e24de677b98409f0c789c
```

```
Package: sipbase, version: 03.12.01.S.154-2.S, status: inactive
  File: bootflash:Active_Dir/asr1000rp2-sipbase.03.12.01.S.154-2.S.pkg, on: SIP0
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: fb815b5cbaf5fd20a0a0e2aeabd2687347c6921d

Package: elcbase, version: 03.12.01.S.154-2.S, status: inactive
  File: bootflash:Active_Dir/asr1000rp2-elcbase.03.12.01.S.154-2.S.pkg, on: SIP0
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: fb1d6abd055b191909bc78ccac23b964de15ab8e

Package: sipspa, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.12.01.S.154-2.S.pkg, on: SIP0/0
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 644364aeea8ccebdd4af5b8d29367db50fc82b17

Package: elcspa, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.12.01.S.154-2.S.pkg, on: SIP0/0
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 2e6b6b1949261873ce5ce189ec19440abffd71c6

Package: sipspa, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.12.01.S.154-2.S.pkg, on: SIP0/1
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 644364aeea8ccebdd4af5b8d29367db50fc82b17

Package: elcspa, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.12.01.S.154-2.S.pkg, on: SIP0/1
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 2e6b6b1949261873ce5ce189ec19440abffd71c6

Package: sipspa, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.12.01.S.154-2.S.pkg, on: SIP0/2
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 644364aeea8ccebdd4af5b8d29367db50fc82b17

Package: elcspa, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.12.01.S.154-2.S.pkg, on: SIP0/2
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 2e6b6b1949261873ce5ce189ec19440abffd71c6

Package: sipspa, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.12.01.S.154-2.S.pkg, on: SIP0/3
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 644364aeea8ccebdd4af5b8d29367db50fc82b17

Package: elcspa, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.12.01.S.154-2.S.pkg, on: SIP0/3
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 2e6b6b1949261873ce5ce189ec19440abffd71c6

Package: sipbase, version: 03.12.01.S.154-2.S, status: inactive
  File: bootflash:Active_Dir/asr1000rp2-sipbase.03.12.01.S.154-2.S.pkg, on: SIP1
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: fb815b5cbaf5fd20a0a0e2aeabd2687347c6921d

Package: elcbase, version: 03.12.01.S.154-2.S, status: inactive
  File: bootflash:Active_Dir/asr1000rp2-elcbase.03.12.01.S.154-2.S.pkg, on: SIP1
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: fb1d6abd055b191909bc78ccac23b964de15ab8e

Package: sipspa, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.12.01.S.154-2.S.pkg, on: SIP1/0
  Built: 2013-03-25_17.28, by: mcpre
```

```
  File SHA1 checksum: 644364aeea8ccebdd4af5b8d29367db50fc82b17

Package: elcspa, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.12.01.S.154-2.S.pkg, on: SIP1/0
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 2e6b6b1949261873ce5ce189ec19440abffd71c6

Package: sipspa, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.12.01.S.154-2.S.pkg, on: SIP1/1
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 644364aeea8ccebdd4af5b8d29367db50fc82b17

Package: elcspa, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.12.01.S.154-2.S.pkg, on: SIP1/1
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 2e6b6b1949261873ce5ce189ec19440abffd71c6

Package: sipspa, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.12.01.S.154-2.S.pkg, on: SIP1/2
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 644364aeea8ccebdd4af5b8d29367db50fc82b17

Package: elcspa, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.12.01.S.154-2.S.pkg, on: SIP1/2
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 2e6b6b1949261873ce5ce189ec19440abffd71c6

Package: sipspa, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.12.01.S.154-2.S.pkg, on: SIP1/3
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 644364aeea8ccebdd4af5b8d29367db50fc82b17

Package: elcspa, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.12.01.S.154-2.S.pkg, on: SIP1/3
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 2e6b6b1949261873ce5ce189ec19440abffd71c6

Package: sipbase, version: 03.13.00.S.154-3.S-ext, status: active
  File: bootflash:Active_Dir/asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg, on: SIP2
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 3b6a4838972840a995ff22e73fd2bae910b268a7

Package: elcbase, version: 03.12.01.S.154-2.S, status: inactive
  File: bootflash:Active_Dir/asr1000rp2-elcbase.03.12.01.S.154-2.S.pkg, on: SIP2
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: fb1d6abd055b191909bc78ccac23b964de15ab8e

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: active
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP2/0
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.12.01.S.154-2.S, status: inactive
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.12.01.S.154-2.S.pkg, on: SIP2/0
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 2e6b6b1949261873ce5ce189ec19440abffd71c6

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: active
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP2/1
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.12.01.S.154-2.S, status: inactive
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.12.01.S.154-2.S.pkg, on: SIP2/1
```

```
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 2e6b6b1949261873ce5ce189ec19440abffd71c6

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: active
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP2/2
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.12.01.S.154-2.S, status: inactive
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.12.01.S.154-2.S.pkg, on: SIP2/2
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 2e6b6b1949261873ce5ce189ec19440abffd71c6

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: active
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP2/3
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.12.01.S.154-2.S, status: inactive
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.12.01.S.154-2.S.pkg, on: SIP2/3
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 2e6b6b1949261873ce5ce189ec19440abffd71c6

Package: sipbase, version: 03.12.01.S.154-2.S, status: inactive
  File: bootflash:Active_Dir/asr1000rp2-sipbase.03.12.01.S.154-2.S.pkg, on: SIP3
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: fb815b5cbaf5fd20a0a0e2aeabd2687347c6921d

Package: elcbase, version: 03.12.01.S.154-2.S, status: inactive
  File: bootflash:Active_Dir/asr1000rp2-elcbase.03.12.01.S.154-2.S.pkg, on: SIP3
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: fb1d6abd055b191909bc78ccac23b964de15ab8e

Package: sipspa, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.12.01.S.154-2.S.pkg, on: SIP3/0
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 644364aeea8ccebdd4af5b8d29367db50fc82b17

Package: elcspa, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.12.01.S.154-2.S.pkg, on: SIP3/0
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 2e6b6b1949261873ce5ce189ec19440abffd71c6

Package: sipspa, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.12.01.S.154-2.S.pkg, on: SIP3/1
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 644364aeea8ccebdd4af5b8d29367db50fc82b17

Package: elcspa, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.12.01.S.154-2.S.pkg, on: SIP3/1
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 2e6b6b1949261873ce5ce189ec19440abffd71c6

Package: sipspa, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.12.01.S.154-2.S.pkg, on: SIP3/2
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 644364aeea8ccebdd4af5b8d29367db50fc82b17

Package: elcspa, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.12.01.S.154-2.S.pkg, on: SIP3/2
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 2e6b6b1949261873ce5ce189ec19440abffd71c6

Package: sipspa, version: 03.12.01.S.154-2.S, status: n/a
```

```
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.12.01.S.154-2.S.pkg, on: SIP3/3
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 644364aeea8ccebdd4af5b8d29367db50fc82b17

Package: elcspa, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.12.01.S.154-2.S.pkg, on: SIP3/3
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 2e6b6b1949261873ce5ce189ec19440abffd71c6

Package: sipbase, version: 03.12.01.S.154-2.S, status: inactive
  File: bootflash:Active_Dir/asr1000rp2-sipbase.03.12.01.S.154-2.S.pkg, on: SIP4
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: fb815b5cbaf5fd20a0a0e2aeabd2687347c6921d

Package: elcbase, version: 03.13.00.S.154-3.S-ext, status: active
  File: bootflash:Active_Dir/asr1000rp2-elcbase.03.13.00.S.154-3.S-ext.pkg, on: SIP4
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 99f8dc925083b118626a4e82d93079050db96826

Package: sipspa, version: 03.12.01.S.154-2.S, status: inactive
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.12.01.S.154-2.S.pkg, on: SIP4/0
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 644364aeea8ccebdd4af5b8d29367db50fc82b17

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: active
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP4/0
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipspa, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.12.01.S.154-2.S.pkg, on: SIP4/1
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 644364aeea8ccebdd4af5b8d29367db50fc82b17

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP4/1
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipspa, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.12.01.S.154-2.S.pkg, on: SIP4/2
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 644364aeea8ccebdd4af5b8d29367db50fc82b17

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP4/2
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipspa, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.12.01.S.154-2.S.pkg, on: SIP4/3
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 644364aeea8ccebdd4af5b8d29367db50fc82b17

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP4/3
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipbase, version: 03.12.01.S.154-2.S, status: inactive
  File: bootflash:Active_Dir/asr1000rp2-sipbase.03.12.01.S.154-2.S.pkg, on: SIP5
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: fb815b5cbaf5fd20a0a0e2aeabd2687347c6921d
```

```
Package: elcbase, version: 03.12.01.S.154-2.S, status: inactive
  File: bootflash:Active_Dir/asr1000rp2-elcbase.03.12.01.S.154-2.S.pkg, on: SIP5
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: fb1d6abd055b191909bc78ccac23b964de15ab8e

Package: sipspa, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.12.01.S.154-2.S.pkg, on: SIP5/0
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 644364aeea8ccebdd4af5b8d29367db50fc82b17

Package: elcspa, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.12.01.S.154-2.S.pkg, on: SIP5/0
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 2e6b6b1949261873ce5ce189ec19440abffd71c6

Package: sipspa, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.12.01.S.154-2.S.pkg, on: SIP5/1
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 644364aeea8ccebdd4af5b8d29367db50fc82b17

Package: elcspa, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.12.01.S.154-2.S.pkg, on: SIP5/1
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 2e6b6b1949261873ce5ce189ec19440abffd71c6

Package: sipspa, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.12.01.S.154-2.S.pkg, on: SIP5/2
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 644364aeea8ccebdd4af5b8d29367db50fc82b17

Package: elcspa, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.12.01.S.154-2.S.pkg, on: SIP5/2
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 2e6b6b1949261873ce5ce189ec19440abffd71c6

Package: sipspa, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.12.01.S.154-2.S.pkg, on: SIP5/3
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 644364aeea8ccebdd4af5b8d29367db50fc82b17

Package: elcspa, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.12.01.S.154-2.S.pkg, on: SIP5/3
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 2e6b6b1949261873ce5ce189ec19440abffd71c6


Router#
Router# redundancy force-switchover
Proceed with switchover to standby RP? [confirm]
<output removed for brevity>
Router#
Router# request platform software package clean
Cleaning up unnecessary package files
No path specified, will use booted path bootflash:Active_Dir/packages.conf
Cleaning bootflash:Active_Dir
  Scanning boot directory for packages ... done.
  Preparing packages list to delete ...
    asr1000rp2-elcbase.03.13.00.S.154-3.S-ext.pkg
      File is in use, will not delete.
    asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg
      File is in use, will not delete.
    asr1000rp2-espbase.03.13.00.S.154-3.S-ext.pkg
      File is in use, will not delete.
    asr1000rp2-espx86base.03.13.00.S.154-3.S-ext.pkg
```

```
      File is in use, will not delete.
    asr1000rp2-rpaccess.03.13.00.S.154-3.S-ext.pkg
      File is in use, will not delete.
    asr1000rp2-rpbase.03.13.00.S.154-3.S-ext.pkg
      File is in use, will not delete.
    asr1000rp2-rpcontrol.03.13.00.S.154-3.S-ext.pkg
      File is in use, will not delete.
    asr1000rp2-rpios-adventerprisek9.03.13.00.S.154-3.S-ext.pkg
      File is in use, will not delete.
    asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg
      File is in use, will not delete.
    asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg
      File is in use, will not delete.
    packages.conf
      File is in use, will not delete.
  done.

Files that will be deleted:
  asr1000rp2-elcbase.03.12.01.S.154-2.S.pkg
  asr1000rp2-elcspa.03.12.01.S.154-2.S.pkg
  asr1000rp2-espbase.03.12.01.S.154-2.S.pkg
  asr1000rp2-espx86base.03.12.01.S.154-2.S.pkg
  asr1000rp2-packages-adventerprisek9.03.12.01.S.154-2.S.conf
  asr1000rp2-rpaccess.03.12.01.S.154-2.S.pkg
  asr1000rp2-rpbase.03.12.01.S.154-2.S.pkg
  asr1000rp2-rpcontrol.03.12.01.S.154-2.S.pkg
  asr1000rp2-rpios-adventerprisek9.03.12.01.S.154-2.S.pkg
  asr1000rp2-sipbase.03.12.01.S.154-2.S.pkg
  asr1000rp2-sipspa.03.12.01.S.154-2.S.pkg
  packages.conf.00-

Do you want to proceed? [confirm]y
  Deleting file bootflash:Active_Dir/asr1000rp2-elcbase.03.12.01.S.154-2.S.pkg ... done.
  Deleting file bootflash:Active_Dir/asr1000rp2-elcspa.03.12.01.S.154-2.S.pkg ... done.
  Deleting file bootflash:Active_Dir/asr1000rp2-espbase.03.12.01.S.154-2.S.pkg ... done.
  Deleting file bootflash:Active_Dir/asr1000rp2-espx86base.03.12.01.S.154-2.S.pkg ...
done.
  Deleting file
bootflash:Active_Dir/asr1000rp2-packages-adventerprisek9.03.12.01.S.154-2.S.conf ... done.
  Deleting file bootflash:Active_Dir/asr1000rp2-rpaccess.03.12.01.S.154-2.S.pkg ... done.
  Deleting file bootflash:Active_Dir/asr1000rp2-rpbase.03.12.01.S.154-2.S.pkg ... done.
  Deleting file bootflash:Active_Dir/asr1000rp2-rpcontrol.03.12.01.S.154-2.S.pkg ... done.
  Deleting file
bootflash:Active_Dir/asr1000rp2-rpios-adventerprisek9.03.12.01.S.154-2.S.pkg ... done.
  Deleting file bootflash:Active_Dir/asr1000rp2-sipbase.03.12.01.S.154-2.S.pkg ... done.
  Deleting file bootflash:Active_Dir/asr1000rp2-sipspa.03.12.01.S.154-2.S.pkg ... done.
  Deleting file bootflash:Active_Dir/packages.conf.00- ... done.
SUCCESS: Files deleted.
```

Using ISSU to Upgrade Subpackages on a Cisco ASR 1006 Router or Cisco ASR 1013 Router (request platform command set) with MDR

This procedure can only be performed if the current ASR 1006 router or ASR 1013 router has two active RPs and both RPs are running subpackages.

To perform an ISSU upgrade with MDR using subpackages on a Cisco ASR 1006 router or ASR 1013 router with a dual RP setup using the **request platform** command set, follow the following instructions.

SUMMARY STEPS

1. **show version**

   show version *active-rp* **installed**

   show version *standby-rp* **installed**

   dir *filesystem*:<*directory*>

   show platform
2. **mkdir** *URL-to-directory-name*
3. **ip tftp** *source-interface gigabitethernet port*
4. **copy tftp:** *URL-to-target-location*
5. **request platform software package expand file** *URL-to-consolidated-package*
6. **dir** *URL-to-consolidated-package*
7. **copy** *file-system*:**asr1000rp2-espbase.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*

   copy *file-system*:**asr1000rp2-espx86base.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*

   copy *file-system*:**asr1000rp2-rpaccess.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*

   copy *file-system*:**asr1000rp2-rpbase.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*

   copy *file-system*:**asr1000rp2-rpcontrol.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*

   copy *file-system*:**asr1000rp2-rpios.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*

   copy *file-system*:**asr1000rp2-sipbase.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*

   copy *file-system*:**asr1000rp2-sipspa.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*

   copy *file-system*:**asr1000rp2-elcbase.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*

   copy *file-system*:**asr1000rp2-elcspa.***version*.**pkg** *URL-to-directory-of-sub-packages-active-RP*
8. **copy** *file-system*:**asr1000rp2-espbase.***version*.**pkg** *URL-to-directory-of-sub-packages-standby-RP*

   copy *file-system*:**asr1000rp2-espx86base.***version*.**pkg** *URL-to-directory-of-sub-packages-standby-RP*

   copy *file-system*:**asr1000rp2-rpaccess.***version*.**pkg** *URL-to-directory-of-sub-packages-standby-RP*

   copy *file-system*:**asr1000rp2-rpbase.***version*.**pkg** *URL-to-directory-of-sub-packages-standby-RP*

   copy *file-system*:**asr1000rp2-rpcontrol.***version*.**pkg** *URL-to-directory-of-sub-packages-standby-RP*

   copy *file-system*:**asr1000rp2-rpios.***version*.**pkg** *URL-to-directory-of-sub-packages-standby-RP*

   copy *file-system*:**asr1000rp2-sipbase.***version*.**pkg** *URL-to-directory-of-sub-packages-standby-RP*

   copy *file-system*:**asr1000rp2-sipspa.***version*.**pkg** *URL-to-directory-of-sub-packages-standby-RP*

   copy *file-system*:**asr1000rp2-elcbase.***version*.**pkg** *URL-to-directory-of-sub-packages-standby-RP*

   copy *file-system*:**asr1000rp2-elcspa.***version*.**pkg** *URL-to-directory-of-sub-packages-standby-RP*
9. **request platform software package verify rp** *slot* **file** *URL* **mdr** {**force**}
10. **request platform software package install rp** *standby-RP* **file** *URL-to-standby-file-system*:**asr1000rp*****version****.pkg force**
11. **hw-module slot** *standby-RP* **reload**

12. **request platform software package install rp** *active-RP* **file** *URL-to-active-file-system*:*image* **slot** *SIP-slot-number* **mdr {force}**

Note Repeat this step for each SIP installed in the router before moving onto the next step.

13. **request platform software package install rp** *active-RP* **file** *URL-to-active-file-system*:*image* **slot** *ELC-slot-number* **mdr {force}**

Note Repeat this step for each ELC installed in the router before moving to the next step.

14. **request platform software package install rp** *active-RP* **file** *URL-to-active-file-system*:**asr1000rp2-esp***version****.pkg slot** *standby-ESP-slot*
15. **request platform software package install rp** *active-RP* **file** *URL-to-active-file-system*:**asr1000rp***version****.pkg force**
16. **show version** *active-RP* **provisioned**

    show version *active-RP* **installed**
17. **redundancy force-switchover**
18. **request platform software package clean**

DETAILED STEPS

	Command or Action	Purpose

Step 1	**show version**
	show version *active-rp* **installed**
	show version *standby-rp* **installed**
	dir *filesystem*:*<directory>*
	show platform
	
	Example:
	`Router# show version`
	`Router# show version r0 installed`
	`Router# show version r1 installed`
	`Router# dir bootflash:`
	`Router# show platform`
		(Optional) Use the following commands to confirm the current router configuration, as follows:
		• **show version** and **show version** *active-rp* **installed**—Verify the running version of the Cisco IOS XE software on the router, and which file was used to boot the router, and where that file is stored.
		• **dir**—Confirm that the files that were used to boot the router are located in the directory.
		• **show platform**—Confirm the current status of the active and standby RPs.

Step 2	**mkdir** *URL-to-directory-name* **Example:** `Router# mkdir bootflash:tmp`	Creates a directory to store the consolidated package and subpackages. This directory must be created in most cases because the consolidated packages and subpackages have to be separated from the subpackages that booted the router at this step of the procedure.
Step 3	**ip tftp source-interface gigabitethernet** *port* **Example:** `Router(config)# ip tftp source-interface gigabitethernet 0`	Specifies the Gigabit Ethernet TFTP source-interface to be configured: *slot/port*—Specifies the location of the TFTP source-interface. **Note** To copy a file using TFTP through the Management Ethernet interface, the **ip tftp source-interface GigabitEthernet 0** command must be entered before entering the **copy tftp** command.
Step 4	**copy tftp:** *URL-to-target-location* **Example:** `Router# copy tftp: bootflash:tmp`	Copies the consolidated package file into the directory created in Step 2. The consolidated package in this step should not be copied into the same directory where the subpackages that are currently running your router are stored (the directory containing the packages.conf provisioning file from which the router was booted). **Tip** It is recommended that you copy the package onto a usb: or harddisk: file system for space considerations when performing this step of the procedure.

	Command or Action	Purpose
Step 5	**request platform software package expand file** *URL-to-consolidated-package* **Example:** `Router# request platform software package expand file bootflash:tmp/asr1000rp2-adventerprisek9.03.13.00.S.154-3.S-ext.bin`	Extracts the subpackages out of the consolidated package file into the temporary directory. **Note** Take extra care to extract the subpackages to a temporary subdirectory and do not delete any of the files currently running the router at this point of the procedure. To erase the files that were running on the router before the ISSU upgrade, enter the **request platform software package clean** command after the ISSU upgrade has been completed.
Step 6	**dir** *target-URL* **Example:** `Router# dir bootflash:tmp`	(Optional) Displays the directory to confirm that the files were extracted.

Command or Action	Purpose

Step 7 **copy** *file-system*:**asr1000rp2-espbase.**_version_.**pkg** *URL-to-directory-of-sub-packages-active-RP*
copy *file-system*:**asr1000rp2-espx86base.**_version_.**pkg** *URL-to-directory-of-sub-packages-active-RP*
copy *file-system*:**asr1000rp2-rpaccess.**_version_.**pkg** *URL-to-directory-of-sub-packages-active-RP*
copy *file-system*:**asr1000rp2-rpbase.**_version_.**pkg** *URL-to-directory-of-sub-packages-active-RP*
copy *file-system*:**asr1000rp2-rpcontrol.**_version_.**pkg** *URL-to-directory-of-sub-packages-active-RP*
copy *file-system*:**asr1000rp2-rpios.**_version_.**pkg** *URL-to-directory-of-sub-packages-active-RP*
copy *file-system*:**asr1000rp2-sipbase.**_version_.**pkg** *URL-to-directory-of-sub-packages-active-RP*
copy *file-system*:**asr1000rp2-sipspa.**_version_.**pkg** *URL-to-directory-of-sub-packages-active-RP*
copy *file-system*:**asr1000rp2-elcbase.**_version_.**pkg** *URL-to-directory-of-sub-packages-active-RP*
copy *file-system*:**asr1000rp2-elcspa.**_version_.**pkg** *URL-to-directory-of-sub-packages-active-RP*
 	
Example:
`Router# copy bootflash:tmp/asr1000rp2-espbase.03.13.00.S.154-3.S-ext.pkg bootflash:`
 	
`Router# copy bootflash:tmp/asr1000rp2-espx86base.03.13.00.S.154-3.S-ext.pkg bootflash:`
 	
`Router# copy bootflash:tmp/asr1000rp2-rpaccess.03.13.00.S.154-3.S-ext.pkg bootflash:`
 	
`Router# copy bootflash:tmp/asr1000rp2-rpbase.03.13.00.S.154-3.S-ext.pkg bootflash:`
 	
`Router# copy bootflash:tmp/asr1000rp2-rpcontrol.03.13.00.S.154-3.S-ext.pkg bootflash:`
 	
`Router# copy bootflash:tmp/asr1000rp2-rpios-adventerprisek9.03.12.01.S.154-2.S1.pkg bootflash:`
 	
`Router# copy bootflash:tmp/asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg bootflash:`
 	
`Router# copy bootflash:tmp/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg bootflash:`
 	
`Router# copy bootflash:tmp/asr1000rp2-elcbase.03.13.00.S.154-3.S-ext.pkg bootflash:`
 	
`Router# copy bootflash:tmp/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg bootflash:`
	Copies the subpackages out of the temporary directory into the directory on the router where the subpackages running the active RP are currently stored.

Command or Action	Purpose

Step 8 **copy** *file-system*:**asr1000rp2-espbase.**_version_.**pkg** *URL-to-directory-of-sub-packages-standby-RP*
copy *file-system*:**asr1000rp2-espx86base.**_version_.**pkg** *URL-to-directory-of-sub-packages-standby-RP*
copy *file-system*:**asr1000rp2-rpaccess.**_version_.**pkg** *URL-to-directory-of-sub-packages-standby-RP*
copy *file-system*:**asr1000rp2-rpbase.**_version_.**pkg** *URL-to-directory-of-sub-packages-standby-RP*
copy *file-system*:**asr1000rp2-rpcontrol.**_version_.**pkg** *URL-to-directory-of-sub-packages-standby-RP*
copy *file-system*:**asr1000rp2-rpios.**_version_.**pkg** *URL-to-directory-of-sub-packages-standby-RP*
copy *file-system*:**asr1000rp2-sipbase.**_version_.**pkg** *URL-to-directory-of-sub-packages-standby-RP*
copy *file-system*:**asr1000rp2-sipspa.**_version_.**pkg** *URL-to-directory-of-sub-packages-standby-RP*
copy *file-system*:**asr1000rp2-elcbase.**_version_.**pkg** *URL-to-directory-of-sub-packages-standby-RP*
copy *file-system*:**asr1000rp2-elcspa.**_version_.**pkg** *URL-to-directory-of-sub-packages-standby-RP*
 	
Example:
`Router# copy bootflash:tmp/asr1000rp2-espbase.03.13.00.S.154-3.S-ext.pkg stby-bootflash:`
 	
`Router# copy bootflash:tmp/asr1000rp2-espx86base.03.13.00.S.154-3.S-ext.pkg stby-bootflash:`
 	
`Router# copy bootflash:tmp/asr1000rp2-rpaccess.03.13.00.S.154-3.S-ext.pkg stby-bootflash:`
 	
`Router# copy bootflash:tmp/asr1000rp2-rpbase.03.13.00.S.154-3.S-ext.pkg stby-bootflash:`
 	
`Router# ccopy bootflash:tmp/asr1000rp2-rpcontrol.03.13.00.S.154-3.S-ext.pkg stby-bootflash:`
 	
`Router# copy bootflash:tmp/asr1000rp2-rpios-adventerprisek9.03.13.00.S.154-3.S-ext.pkg`
 	
`Router# copy bootflash:tmp/asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg stby-bootflash:`
 	
`Router# copy bootflash:tmp/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg stby-bootflash:`
 	
`Router# copy bootflash:tmp/asr1000rp2-elcbase.03.13.00.S.154-3.S-ext.pkg stby-bootflash:`
 	
`Router# copy bootflash:tmp/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg stby-bootflash:`
	Copies the subpackages out of the temporary directory into the directory on the router where the subpackages running the standby RP are currently stored.

	Command or Action	Purpose
Step 9	**request platform software package verify rp** *slot* **file** *URL* **mdr** {**force**} **Example:** `Router# request platform software package verify rp 1 file stby-bootflash:asr1000rp2-*03.13.00.S.154-3.S-ext*.pkg mdr`	Verifies the RP subpackages on the standby RP, where the "**rp***" wildcard is specified to capture all of the RP subpackages for the desired upgrade release.
Step 10	**request platform software package install rp** *standby-RP* **file** *target-standbyRP-URL-for-sub-packages*`:asr1000rp*`*version*`*.pkg force` **Example:** `Router# request platform software package install rp 1 file stby-bootflash:asr1000rp2-*03.13.00.S.154-3.S-ext*.pkg force`	Upgrades the RP subpackages on the standby RP, where the "**rp***" wildcard is specified to capture all of the RP subpackages for the desired upgrade release.
Step 11	**hw-module slot** *standby-RP* **reload** **Example:** `Router# hw-module slot R1 reload`	Reloads the standby RP.
Step 12	**request platform software package install rp** *active-RP* **file** *URL-to-active-file-system*`:asr1000rp2-{sipbase,sipspa}*`*version*`*.pkg slot` *SIP-slot-number* **mdr** {**force**} **Example:** `Router# request platform software package install rp 0 file bootflash:asr1000rp2-{sipbase,sipspa}*03.13.00.S.154-3*pkg mdr`	Upgrades the SIP and SPA subpackages for each SIP on the router using MDR. **Note** This step must be completed one SIP at a time, and repeated for each SIP installed on the router before performing the next step. **Tip** You can use the **show ip interface brief** command to identify which slots contain SIPs and SPAs. The interfaces with three numbers (in the form *SIP-number*/*SPA-number*/*interface-number*) identify the SIP and SPA locations in the router. **Note** The *pattern* options used in this CLI (**sipbase** and **sipspa**) were introduced in Cisco IOS XE Release 2.1.2 and are not available in previous Cisco IOS XE Releases. See the "ISSU Procedures (Prior to Cisco IOS XE Release 2.1.2)" section on page 5-72 for pre-Cisco IOS XE Release 2.1.2 ISSU upgrade procedures.

	Command or Action	Purpose
Step 13	**request platform software package install rp** *active-RP* **file** *URL-to-active-file-system*:**asr1000rp2-{elcbase,elcspa}***version****.pkg slot** *ELC-slot-number* **mdr {force}** **Example:** `Router# request platform software package install rp 0 file bootflash:asr1000rp2-{elcbase,elcspa}*03.13.00.S.154-3*pkg mdr`	Upgrades the ELC and SPA subpackages for each ELC on the router using MDR. **Note** This step must be completed for one ELC at a time, and repeated for each ELC installed on the router before performing the next step. **Tip** You can use the **show ip interface brief** command to identify which slots contain ELCs and SPAs. The interfaces with three numbers (in the form *ELC-number*/*SPA-number*/*interface-number*) identify the ELC and SPA locations in the router. **Note** The *pattern* options used in this CLI (**elcbase** and **elcspa**) were introduced in Cisco IOS XE Release 3.10S and are not available in previous Cisco IOS XE Releases.

Step 14	**request platform software package install rp** *active-RP* **file** *URL-to-active-file-system*:**asr1000rp2-esp***version****.pkg slot** *standby-ESP-slot*
	
	request platform software package install rp *active-RP* **file** *URL-to-active-file-system*:**asr1000rp2-esp***version****.pkg slot** *active-ESP-slot*
	
	Example:
	`Router# request platform software package install rp 0 file bootflash:asr1000rp2-esp*03.13.00.S.154-3.S-ext*pkg slot 1`
	
	`Router# request platform software package install rp 0 file bootflash:asr1000rp2-esp*03.13.00.S.154-3.S-ext*pkg slot 0`
		Upgrades the ESP Base subpackage on the standby and the active ESPs.
		After entering the **issu loadversion rp** command on the active RP, the ESP switchover will occur automatically. Minimal traffic interruption will occur as a result of this switchover.

Step 15	**request platform software package install rp** *active-RP* **file** *URL-to-active-file-system*:**asr1000rp***version****.pkg force** **Example:** `Router# request platform software package install rp 0 file bootflash:asr1000rp2-*03.13.00.S.154-3.S-ext*.pkg force`	Upgrades all of the sub-packages on the active RP. **Note** This step is required to ensure that all subpackages on the router were upgraded as part of this procedure, and might upgrade some subpackages that would otherwise be missed in the process.

Step 16	**show version** *active-RP* **provisioned**
	show version *active-RP* **installed**
	
	Example:
	`Router# show version r0 provisioned`
	`Router# show version r0 installed`
		(Optional) Confirms the subpackages are provisioned and installed.

	Command or Action	Purpose
Step 17	**redundancy force-switchover** **Example:** `Router# redundancy force-switchover`	Forces an RP switchover to complete the upgrade.
Step 18	**request platform software package clean**	(Optional) Removes all unused subpackage files from the router.

Examples

This example shows ISSU upgrade using subpackages on a Cisco ASR 1006 router or ASR 1013 router with a dual RP setup:

```
Router# show version
Cisco IOS Software, IOS-XE Software (X86_64_LINUX_IOSD-ADVENTERPRISEK9-M), Version
15.3(2)S, RELEASE SOFTWARE (fc1)

<output removed for brevity>
System image file is "bootflash:Active_Dir/packages.conf"

<output removed for brevity>
cisco ASR1013 (RP2) processor with 4208889K/6147K bytes of memory.
Processor board ID FOX1343GJGC
20 Gigabit Ethernet interfaces
6 Ten Gigabit Ethernet interfaces
32768K bytes of non-volatile configuration memory.
8388608K bytes of physical memory.
1925119K bytes of eUSB flash at bootflash:.
78085207K bytes of SATA hard disk at harddisk:.

Configuration register is 0x2102

Router# show platform
Chassis type: ASR1013

Slot      Type                State                 Insert time (ago)
--------- ------------------- --------------------- -----------------
 2        ASR1000-SIP40         ok                    1d03h
 2/0      SPA-1X10GE-L-V2       ok                    1d03h
 2/1      SPA-1X10GE-L-V2       ok                    1d03h
 2/2      SPA-1X10GE-L-V2       ok                    1d03h
 2/3      SPA-1X10GE-L-V2       ok                    1d03h
 4        ASR1000-2T+20X1GE     ok                    1d03h
 4/0      BUILT-IN-2T+20X1GE    ok                    1d03h
 R0       ASR1000-RP2           ok, active            1d03h
 R1       ASR1000-RP2           ok, standby           1d03h
 F0       ASR1000-ESP100        ok, active            1d03h
 F1       ASR1000-ESP100        ok, standby           1d03h
 P0       ASR1013-PWR-AC        ok                    1d03h
 P1       ASR1013-PWR-AC        ok                    1d03h
 P2       ASR1013-PWR-AC        ok                    1d03h
 P3       ASR1013-PWR-AC        ps, fail              1d03h

Slot      CPLD Version        Firmware Version
--------- ------------------- ---------------------------------------
2         00200800            15.3(3r)S
4         00200800            15.3(1r)S
R0        10021901            15.3(3r)S
R1        10021901            15.3(3r)S
F0        12071700            15.3(3r)S
```

```
F1        12071700            15.3(3r)S

Router# show version r0 installed
Package: Provisioning File, version: n/a, status: active
  File: bootflash:Active_Dir/packages.conf, on: RP0
  Built: n/a, by: n/a
  File SHA1 checksum: a624f70f68c60292f4482433f43afd92487a55c4

Package: rpbase, version: 03.12.01.S.154-2.S, status: active
  File: bootflash:Active_Dir/asr1000rp2-rpbase.03.12.01.S.154-2.S.pkg, on: RP0
  Built: 2013-03-25_18.48, by: mcpre
  File SHA1 checksum: 3a9675142898cfac350d4e42f0e37bd9f4e48538

Package: rpcontrol, version: 03.12.01.S.154-2.S, status: active
  File: bootflash:Active_Dir/asr1000rp2-rpcontrol.03.12.01.S.154-2.S.pkg, on: RP0/0
  Built: 2013-03-25_18.48, by: mcpre
  File SHA1 checksum: 87b11f863f67fdf2610ee0769b929baab4c3efad
<output removed for brevity>
Router# dir bootflash:Active_Dir
Directory of bootflash:/Active_Dir/

   20  -rw-    41104112   Aug 3 2013 15:05:40 +05:30
asr1000rp2-elcbase.03.12.01.S.154-2.S.pkg
   21  -rw-    50285296   Aug 3 2013 15:05:40 +05:30
asr1000rp2-elcspa.03.12.01.S.154-2.S.pkg
   22  -rw-    82514676   Aug 3 2013 15:05:40 +05:30
asr1000rp2-espbase.03.12.01.S.154-2.S.pkg
   23  -rw-   101084628   Aug 3 2013 15:05:40 +05:30
asr1000rp2-espx86base.03.12.01.S.154-2.S.pkg
   17  -rw-        9059   Aug 3 2013 15:05:40 +05:30
asr1000rp2-packages-adventerprisek9.03.12.01.S.154-2.S.conf
   24  -rw-    29012724   Aug 3 2013 15:05:40 +05:30
asr1000rp2-rpaccess.03.12.01.S.154-2.S.pkg
   25  -rw-    49898964   Aug 3 2013 15:05:40 +05:30
asr1000rp2-rpbase.03.12.01.S.154-2.S.pkg
   26  -rw-    46557940   Aug 3 2013 15:05:40 +05:30
asr1000rp2-rpcontrol.03.12.01.S.154-2.S.pkg
   27  -rw-   114612988   Aug 3 2013 15:05:41 +05:30
asr1000rp2-rpios-adventerprisek9.03.12.01.S.154-2.S.pkg
   28  -rw-    41954036   Aug 3 2013 15:05:41 +05:30
asr1000rp2-sipbase.03.12.01.S.154-2.S.pkg
   29  -rw-    60957428   Aug 3 2013 15:05:41 +05:30
asr1000rp2-sipspa.03.12.01.S.154-2.S.pkg
   19  -rw-        9838   Aug 3 2013 15:05:41 +05:30  packages.conf

1940303872 bytes total (503164928 bytes free)
Router#
Router# show redundancy states
       my state = 13 -ACTIVE
     peer state = 8  -STANDBY HOT
           Mode = Duplex
           Unit = Primary
        Unit ID = 48

Redundancy Mode (Operational) = sso
Redundancy Mode (Configured)  = sso
Redundancy State              = sso
     Maintenance Mode = Disabled
    Manual Swact = enabled
 Communications = Up

   client count = 108
 client_notification_TMR = 30000 milliseconds
           RF debug mask = 0x0
```

```
Router# copy running-config startup-config
Destination filename [startup-config]?
Building configuration...
[OK]
Router# mkdir harddisk:Target_Subs
Create directory filename [Target_Subs]?
Created dir harddisk:/Target_Subs
Router#
Router#
Router#
Router# request platform software package expand file
harddisk:Target_Subs/asr1000rp2-adventerprisek9.03.13.00.S.154-3.S-ext.bin to
harddisk:Target_Subs
Verifying parameters
Validating package type
Copying package files
SUCCESS: Finished expanding all-in-one software package.
Router#
Router#
Router# dir harddisk:Target_Subs
Directory of harddisk:/Target_Subs/

3358722  -rw-   569597380   Aug 4 2013 18:45:38 +05:30
asr1000rp2-adventerprisek9.03.13.00.S.154-3.S-ext.bin
7684099  -rw-    37557200   Aug 4 2013 18:46:43 +05:30
asr1000rp2-elcbase.03.13.00.S.154-3.S-ext.pkg
7684100  -rw-    51194832   Aug 4 2013 18:46:43 +05:30
asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg
7684101  -rw-    80657364   Aug 4 2013 18:46:43 +05:30
asr1000rp2-espbase.03.13.00.S.154-3.S-ext.pkg
7684102  -rw-    95446456   Aug 4 2013 18:46:43 +05:30
asr1000rp2-espx86base.03.13.00.S.154-3.S-ext.pkg
7684097  -rw-        9381   Aug 4 2013 18:46:43 +05:30
asr1000rp2-packages-adventerprisek9.03.13.00.S.154-3.S-ext.conf
7684103  -rw-    23350232   Aug 4 2013 18:46:43 +05:30
asr1000rp2-rpaccess.03.13.00.S.154-3.S-ext.pkg
7684104  -rw-    37694900   Aug 4 2013 18:46:44 +05:30
asr1000rp2-rpbase.03.13.00.S.154-3.S-ext.pkg
7684105  -rw-    45536216   Aug 4 2013 18:46:44 +05:30
asr1000rp2-rpcontrol.03.13.00.S.154-3.S-ext.pkg
7684106  -rw-   118754284   Aug 4 2013 18:46:44 +05:30
asr1000rp2-rpios-adventerprisek9.03.13.00.S.154-3.S-ext.pkg
7684107  -rw-    38380500   Aug 4 2013 18:46:44 +05:30
asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg
7684108  -rw-    61760468   Aug 4 2013 18:46:44 +05:30
asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg
7684098  -rw-       10165   Aug 4 2013 18:46:44 +05:30  packages.conf

78704144384 bytes total (9254879232 bytes free)
Router# copy harddisk:Target_Subs/asr1000rp2-espbase.03.13.00.S.154-3.S-ext.pkg
bootflash:Active_Dir/
Destination filename [Active_Dir/asr1000rp2-espbase.03.13.00.S.154-3.S-ext.pkg]?
Copy in
progress...CCCC
```

```
Copy in
progress...CCCC
```

```
CCCC
```

```
CCCC
```

```
Destination filename [Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg]?
Copy in
progress...CCCC
```

```
    WARNING: In-service installation of IOSD package
    WARNING: requires software redundancy on target RP
    WARNING: or on-reboot parameter
    WARNING: Automatically setting the on-reboot flag
    WARNING: In-service installation of RP Base package
    WARNING: requires software reboot of target RP
Processing image file constraints
Creating candidate provisioning file
Finished image file verification

--- Starting candidate package set construction ---
Verifying existing software set
Processing candidate provisioning file
Constructing working set for candidate package set
Constructing working set for running package set
Checking command output
Constructing merge of running and candidate packages
Checking if resulting candidate package set would be complete
Finished candidate package set construction

--- Starting compatibility testing ---
Determining whether candidate package set is compatible
Determining whether installation is valid
Determining whether installation is valid ... skipped
Verifying image type compatibility
Checking IPC compatibility for candidate software
Checking candidate package set infrastructure compatibility
Checking infrastructure compatibility with running software
Checking infrastructure compatibility with running software ... skipped
Checking package specific compatibility
Finished compatibility testing

--- Starting mdr compatibility verification ---
MDR for CC type [0x762] located at slot [4] not supported by running package version
[03.12.01.S.154-2.S]
As SIP4 does not support MDR none of the SPA's within in may be upgraded using MDR
MDR compatibility failed - proceeding with forced MDR-upgrade - some traffic will be
impacted during the upgrade
Finished mdr compatibility verification

SUCCESS: Software is ISSU MDR compatible.


Router# request platform software package install rp 1 file
stby-bootflash:Active_Dir/asr1000rp*03.13.00.S.154-3.S-ext*.pkg force
--- Starting local lock acquisition on R0 ---
Finished local lock acquisition on R0

--- Starting installation state synchronization ---
Finished installation state synchronization

--- Starting local lock acquisition on R1 ---
Finished local lock acquisition on R1

--- Starting file path checking ---
Finished file path checking

--- Starting image file verification ---
Checking image file names
Locating image files and validating name syntax
  Found asr1000rp2-elcbase.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-espbase.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-espx86base.03.13.00.S.154-3.S-ext.pkg
```

```
  Found asr1000rp2-rpaccess.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-rpbase.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-rpcontrol.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-rpios-adventerprisek9.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg
Verifying image file locations
Inspecting image file types
    WARNING: In-service installation of IOSD package
    WARNING: requires software redundancy on target RP
    WARNING: or on-reboot parameter
    WARNING: Automatically setting the on-reboot flag
    WARNING: In-service installation of RP Base package
    WARNING: requires software reboot of target RP
Processing image file constraints
Creating candidate provisioning file
Finished image file verification

--- Starting candidate package set construction ---
Verifying existing software set
Processing candidate provisioning file
Constructing working set for candidate package set
Constructing working set for running package set
Checking command output
Constructing merge of running and candidate packages
Checking if resulting candidate package set would be complete
Finished candidate package set construction

--- Starting compatibility testing ---
Determining whether candidate package set is compatible
Determining whether installation is valid
Determining whether installation is valid ... skipped
Verifying image type compatibility
Checking IPC compatibility for candidate software
Checking candidate package set infrastructure compatibility
Checking infrastructure compatibility with running software
Checking infrastructure compatibility with running software ... skipped
Checking package specific compatibility
Finished compatibility testing

--- Starting list of software package changes ---
Old files list:
  Removed asr1000rp2-elcbase.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-elcspa.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-espbase.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-espx86base.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-rpaccess.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-rpbase.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-rpcontrol.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-rpios-adventerprisek9.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-sipbase.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-sipspa.03.12.01.S.154-2.S.pkg
New files list:
  Added asr1000rp2-elcbase.03.13.00.S.154-3.S-ext.pkg
  Added asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg
  Added asr1000rp2-espbase.03.13.00.S.154-3.S-ext.pkg
  Added asr1000rp2-espx86base.03.13.00.S.154-3.S-ext.pkg
  Added asr1000rp2-rpaccess.03.13.00.S.154-3.S-ext.pkg
  Added asr1000rp2-rpbase.03.13.00.S.154-3.S-ext.pkg
  Added asr1000rp2-rpcontrol.03.13.00.S.154-3.S-ext.pkg
  Added asr1000rp2-rpios-adventerprisek9.03.13.00.S.154-3.S-ext.pkg
  Added asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg
  Added asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg
Finished list of software package changes
```

```
--- Starting commit of software changes ---
Updating provisioning rollback files
Creating pending provisioning file
Committing provisioning file
Finished commit of software changes

SUCCESS: Software provisioned.  New software will load on reboot.


Router# hw-module slot r1 reload
Proceed with reload of module? [confirm]
Router#
*Aug  4 19:14:01.721 IST: %IOSXE_OIR-6-OFFLINECARD: Card (rp) offline in slot R1
*Aug  4 19:14:01.761 IST: %REDUNDANCY-3-STANDBY_LOST: Standby processor fault
(PEER_NOT_PRESENT)
*Aug  4 19:14:01.761 IST: %REDUNDANCY-3-STANDBY_LOST: Standby processor fault (PEER_DOWN)
*Aug  4 19:14:01.761 IST: %REDUNDANCY-3-STANDBY_LOST: Standby processor fault
(PEER_REDUNDANCY_STATE_CHANGE)
*Aug  4 19:14:03.584 IST: %RF-5-RF_RELOAD: Peer reload. Reason: EHSA standby down
*Aug  4 19:14:03.594 IST: % Redundancy mode change to SSO

Router#
*Aug  4 19:17:35.443 IST: %IOSXE_OIR-6-ONLINECARD: Card (rp) online in slot R1
Router#
*Aug  4 19:17:48.061 IST: %REDUNDANCY-5-PEER_MONITOR_EVENT: Active detected a standby
insertion (raw-event=PEER_FOUND(4))

*Aug  4 19:17:48.061 IST: %REDUNDANCY-5-PEER_MONITOR_EVENT: Active detected a standby
insertion (raw-event=PEER_REDUNDANCY_STATE_CHANGE(5))

*Aug  4 19:17:51.003 IST: %REDUNDANCY-3-IPC: IOS versions do not match.

*Aug  4 19:19:08.380 IST: %NBAR_HA-5-NBAR_INFO: NBAR sync DONE!
*Aug  4 19:19:08.797 IST: %HA_CONFIG_SYNC-6-BULK_CFGSYNC_SUCCEED: Bulk Sync succeeded
*Aug  4 19:19:08.798 IST: %RF-5-RF_TERMINAL_STATE: Terminal state reached for (SSO)
Router#
Router#request platform software package install rp 0 file
bootflash:Active_Dir/asr1000rp2-{sipbase,sipspa}*03.13.00.S.154-3.S-ext*.pkg slot 2 mdr
force
--- Starting local lock acquisition on R0 ---
Finished local lock acquisition on R0

--- Starting installation state synchronization ---
Finished installation state synchronization

--- Starting file path checking ---
Finished file path checking

--- Starting image file verification ---
Checking image file names
Locating image files and validating name syntax
  Found asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg
Verifying image file locations
Inspecting image file types
Processing image file constraints
Creating candidate provisioning file
Finished image file verification

--- Starting candidate package set construction ---
Verifying existing software set
Processing candidate provisioning file
Constructing working set for candidate package set
```

```
Constructing working set for running package set
Checking command output
Constructing merge of running and candidate packages
Checking if resulting candidate package set would be complete
Finished candidate package set construction

--- Starting compatibility testing ---
Determining whether candidate package set is compatible

WARNING:
WARNING: Candidate software combination not found in compatibility database
WARNING:

Determining whether installation is valid
Creating matrix_file by locate_latest_matrix_file  /tmp/issu/provision/sw

WARNING:
WARNING: Candidate software combination not found in compatibility database
WARNING:


WARNING:
WARNING: Candidate software combination not found in compatibility database
WARNING:

Software sets are identified as compatible
Verifying image type compatibility
Checking IPC compatibility with running software
Checking candidate package set infrastructure compatibility
Checking infrastructure compatibility with running software
Checking package specific compatibility
Finished compatibility testing

--- Starting mdr compatibility verification ---
Finished mdr compatibility verification

--- Starting impact testing ---
Checking operational impact of change
Finished impact testing

--- Starting list of software package changes ---
No old package files removed
New files list:
  Added asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg
  Added asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg
Finished list of software package changes

--- Starting commit of software changes ---
Updating provisioning rollback files
Creating pending provisioning file
Committing provisioning file
Finished commit of software changes

--- Starting analysis of software changes ---
Finished analysis of software changes

--- Starting update running software ---
Blocking peer synchronization of operating information
Creating the command set placeholder directory
  Finding latest command set
  Finding latest command shortlist lookup file
  Finding latest command shortlist file
  Assembling CLI output libraries
  Assembling CLI input libraries
```

```
  Assembling Dynamic configuration files
  Applying interim IPC and database definitions
  Replacing running software
  Replacing CLI software
  Restarting software

*Aug  4 19:20:48.949 IST: %MDR-5-CARD_RESTART: R0/0: card_mdr:  Minimal Disruptive Restart
SIP2 to acquire provisioned software  Applying final IPC and database definitions

*Aug  4 19:20:58.013 IST: %CMCC-5-SPA_MDR_INIT: SIP2: cmcc:  SPA0 initiated Minimal
Disruptive Restart
*Aug  4 19:20:58.017 IST: %CMCC-5-SPA_MDR_INIT: SIP2: cmcc:  SPA1 initiated Minimal
Disruptive Restart
*Aug  4 19:20:58.018 IST: %CMCC-5-SPA_MDR_INIT: SIP2: cmcc:  SPA2 initiated Minimal
Disruptive Restart
*Aug  4 19:20:58.019 IST: %CMCC-5-SPA_MDR_INIT: SIP2: cmcc:  SPA3 initiated Minimal
Disruptive Restart  Generating software version information
  Notifying running software of updates
  Unblocking peer synchronization of operating information
Unmounting old packages
Cleaning temporary installation files
  Finished update running software

SUCCESS: Finished installing software.

Router#
*Aug  4 19:21:45.424 IST: %IOSXE_OIR-6-ONLINECARD: Card (cc) online in slot 2
*Aug  4 19:21:48.382 IST: %IOSXE_OIR-6-INSSPA: SPA inserted in subslot 2/0
*Aug  4 19:21:48.733 IST: %IOSXE_OIR-6-INSSPA: SPA inserted in subslot 2/1
*Aug  4 19:21:49.083 IST: %IOSXE_OIR-6-INSSPA: SPA inserted in subslot 2/2
*Aug  4 19:21:49.430 IST: %IOSXE_OIR-6-INSSPA: SPA inserted in subslot 2/3
*Aug  4 19:21:58.121 IST: %LINK-3-UPDOWN: SIP2/0: Interface EOBC2/1, changed state to up
*Aug  4 19:22:02.302 IST: %SPA_OIR-6-ONLINECARD: SPA (SPA-1X10GE-L-V2) online in subslot
2/0
*Aug  4 19:22:02.312 IST: %CMCC-5-SPA_MDR_DONE: SIP2: cmcc:  SPA0 completed Minimal
Disruptive Restart
*Aug  4 19:22:02.282 IST: %TRANSCEIVER-6-INSERTED: SIP2/0: transceiver module inserted in
TenGigabitEthernet2/0/0
*Aug  4 19:22:02.518 IST: %LINK-3-UPDOWN: SIP2/1: Interface EOBC2/1, changed state to up
*Aug  4 19:22:06.113 IST: %SPA_OIR-6-ONLINECARD: SPA (SPA-1X10GE-L-V2) online in subslot
2/1
*Aug  4 19:22:06.126 IST: %CMCC-5-SPA_MDR_DONE: SIP2: cmcc:  SPA1 completed Minimal
Disruptive Restart
*Aug  4 19:22:06.082 IST: %TRANSCEIVER-6-INSERTED: SIP2/1: transceiver module inserted in
TenGigabitEthernet2/1/0
*Aug  4 19:22:08.080 IST: %LINK-3-UPDOWN: SIP2/2: Interface EOBC2/1, changed state to up
*Aug  4 19:22:11.627 IST: %SPA_OIR-6-ONLINECARD: SPA (SPA-1X10GE-L-V2) online in subslot
2/2
*Aug  4 19:22:11.637 IST: %CMCC-5-SPA_MDR_DONE: SIP2: cmcc:  SPA2 completed Minimal
Disruptive Restart
*Aug  4 19:22:11.616 IST: %TRANSCEIVER-6-INSERTED: SIP2/2: transceiver module inserted in
TenGigabitEthernet2/2/0
*Aug  4 19:22:12.523 IST: %LINK-3-UPDOWN: SIP2/3: Interface EOBC2/1, changed state to up
*Aug  4 19:22:16.657 IST: %SPA_OIR-6-ONLINECARD: SPA (SPA-1X10GE-L-V2) online in subslot
2/3
*Aug  4 19:22:16.668 IST: %CMCC-5-SPA_MDR_DONE: SIP2: cmcc:  SPA3 completed Minimal
Disruptive Restart
*Aug  4 19:22:16.648 IST: %TRANSCEIVER-6-INSERTED: SIP2/3: transceiver module inserted in
TenGigabitEthernet2/3/0
Router#
Router#
Router#issu commitversion
--- Starting local lock acquisition on R0 ---
Finished local lock acquisition on R0
```

```
--- Starting installation changes ---
Cancelling rollback timer
Finished installation changes

SUCCESS: Installation changes committed
Router#
Router#
Router#request platform software package install rp 0 file
bootflash:Active_Dir/asr1000rp2-{elcbase,elcspa}*03.13.00.S.154-3.S-ext*.pkg slot 4
--- Starting local lock acquisition on R0 ---
Finished local lock acquisition on R0

--- Starting installation state synchronization ---
Finished installation state synchronization

--- Starting file path checking ---
Finished file path checking

--- Starting image file verification ---
Checking image file names
Locating image files and validating name syntax
  Found asr1000rp2-elcbase.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg
Verifying image file locations
Inspecting image file types
Processing image file constraints
Creating candidate provisioning file
Finished image file verification

--- Starting candidate package set construction ---
Verifying existing software set
Processing candidate provisioning file
Constructing working set for candidate package set
Constructing working set for running package set
Checking command output
Constructing merge of running and candidate packages
Checking if resulting candidate package set would be complete
Finished candidate package set construction

--- Starting compatibility testing ---
Determining whether candidate package set is compatible

WARNING:
WARNING: Candidate software combination not found in compatibility database
WARNING:

Determining whether installation is valid
Creating matrix_file by locate_latest_matrix_file  /tmp/issu/provision/sw

WARNING:
WARNING: Candidate software combination not found in compatibility database
WARNING:


WARNING:
WARNING: Candidate software combination not found in compatibility database
WARNING:

Software sets are identified as compatible
Verifying image type compatibility
Checking IPC compatibility with running software
Checking candidate package set infrastructure compatibility
Checking infrastructure compatibility with running software
```

```
Checking package specific compatibility
Finished compatibility testing

--- Starting impact testing ---
Checking operational impact of change
Finished impact testing

--- Starting list of software package changes ---
No old package files removed
New files list:
  Added asr1000rp2-elcbase.03.13.00.S.154-3.S-ext.pkg
  Added asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg
Finished list of software package changes

--- Starting commit of software changes ---
Updating provisioning rollback files
Creating pending provisioning file
Committing provisioning file
Finished commit of software changes

--- Starting analysis of software changes ---
Finished analysis of software changes

--- Starting update running software ---
Blocking peer synchronization of operating information
Creating the command set placeholder directory
  Finding latest command set
  Finding latest command shortlist lookup file
  Finding latest command shortlist file
  Assembling CLI output libraries
  Assembling CLI input libraries
  Assembling Dynamic configuration files
  Applying interim IPC and database definitions
  Replacing running software
  Replacing CLI software
  Restarting software
  Applying final IPC and database definitions
  *Aug 4 19:41:12.290 IST: %MDR-5-CARD_RESTART: R0/0: card_mdr: Minimal Disruptive Restart
SIP4 to acquire provisioned software
*Aug 4 19:41:22.408 IST: %CMCC-5-SPA_MDR_INIT: SIP4: cmcc: SPA0 initiated Minimal
Disruptive Restart Generating software version information
  Notifying running software of updates
  Unblocking peer synchronization of operating information
Unmounting old packages
Cleaning temporary installation files
  Finished update running software

SUCCESS: Finished installing software.

*Aug 4 19:42:53.778 IST: %IOSXE_OIR-6-ONLINECARD: Card (cc) online in slot 4
*Aug 4 19:42:57.364 IST: %IOSXE_OIR-6-INSSPA: SPA inserted in subslot 4/0
*Aug 4 19:43:17.541 IST: %LINK-3-UPDOWN: SIP4/0: Interface EOBC0/1, changed state to up
*Aug 4 19:43:25.251 IST: %SPA_OIR-6-ONLINECARD: SPA (BUILT-IN-2T+20X1GE) online in subslot
4/0
*Aug 4 19:43:25.291 IST: %CMCC-5-SPA_MDR_DONE: SIP4: cmcc: SPA0 completed Minimal
Disruptive Restart
*Aug 4 19:43:26.815 IST: %LINK-3-UPDOWN: SIP4/0: Interface GigabitEthernet4/0/0, changed
state to up
*Aug 4 19:43:28.267 IST: %LINK-3-UPDOWN: SIP4/0: Interface GigabitEthernet4/0/1, changed
state to up

Router# issu commitversion
--- Starting local lock acquisition on R0 ---
Finished local lock acquisition on R0
```

```
--- Starting installation changes ---
Cancelling rollback timer
Finished installation changes

SUCCESS: Installation changes committed
Router#
Router#
Router# request platform software package install rp 0 file
bootflash:Active_Dir/asr1000rp2-esp*03.13.00.S.154-3.S-ext*.pkg slot 1
--- Starting local lock acquisition on R0 ---
Finished local lock acquisition on R0

--- Starting installation state synchronization ---
Finished installation state synchronization

--- Starting file path checking ---
Finished file path checking

--- Starting image file verification ---
Checking image file names
Locating image files and validating name syntax
  Found asr1000rp2-espbase.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-espx86base.03.13.00.S.154-3.S-ext.pkg
Verifying image file locations
Inspecting image file types
Processing image file constraints
Creating candidate provisioning file
Finished image file verification

--- Starting candidate package set construction ---
Verifying existing software set
Processing candidate provisioning file
Constructing working set for candidate package set
Constructing working set for running package set
Checking command output
Constructing merge of running and candidate packages
Checking if resulting candidate package set would be complete
Finished candidate package set construction

--- Starting compatibility testing ---
Determining whether candidate package set is compatible

WARNING:
WARNING: Candidate software combination not found in compatibility database
WARNING:

Determining whether installation is valid
Creating matrix_file by locate_latest_matrix_file  /tmp/issu/provision/sw

WARNING:
WARNING: Candidate software combination not found in compatibility database
WARNING:


WARNING:
WARNING: Candidate software combination not found in compatibility database
WARNING:

Software sets are identified as compatible
Verifying image type compatibility
Checking IPC compatibility with running software
Checking candidate package set infrastructure compatibility
Checking infrastructure compatibility with running software
```

```
Checking package specific compatibility
Finished compatibility testing

--- Starting impact testing ---
Checking operational impact of change
Finished impact testing

--- Starting list of software package changes ---
No old package files removed
New files list:
  Added asr1000rp2-espbase.03.13.00.S.154-3.S-ext.pkg
  Added asr1000rp2-espx86base.03.13.00.S.154-3.S-ext.pkg
Finished list of software package changes

--- Starting commit of software changes ---
Updating provisioning rollback files
Creating pending provisioning file
Committing provisioning file
Finished commit of software changes

--- Starting analysis of software changes ---
Finished analysis of software changes

--- Starting update running software ---
Blocking peer synchronization of operating information
Creating the command set placeholder directory
  Finding latest command set
  Finding latest command shortlist lookup file
  Finding latest command shortlist file
  Assembling CLI output libraries
  Assembling CLI input libraries
  Assembling Dynamic configuration files
  Applying interim IPC and database definitions
  Replacing running software
  Replacing CLI software
  Restarting software
    Restarting ESP1
  Applying final IPC and database definitions

*Aug  4 19:29:16.751 IST: %IOSXE_OIR-6-OFFLINECARD: Card (fp) offline in slot F1
*Aug  4 19:29:18.172 IST: %CMRP-6-FP_HA_STATUS: R0/0: cmand:  F0 redundancy state is
Active with no Standby  Generating software version information
  Notifying running software of updates
  Unblocking peer synchronization of operating information
Unmounting old packages
Cleaning temporary installation files
  Finished update running software

SUCCESS: Finished installing software.

Router#
*Aug  4 19:30:50.972 IST: %CPPHA-7-START: F1: cpp_ha:  CPP 0 preparing image
/tmp/sw/fp/1/0/fpx86/mount/usr/cpp/bin/qfp-ucode-esp40
*Aug  4 19:30:51.362 IST: %CPPHA-7-START: F1: cpp_ha:  CPP 0 startup init image
/tmp/sw/fp/1/0/fpx86/mount/usr/cpp/bin/qfp-ucode-esp40
*Aug  4 19:30:53.088 IST: %IOSXE_OIR-6-ONLINECARD: Card (fp) online in slot F1
Router#
Router# issu commitversion
--- Starting local lock acquisition on R0 ---
Finished local lock acquisition on R0

--- Starting installation changes ---
Cancelling rollback timer
Finished installation changes
```

```
SUCCESS: Installation changes committed
Router#
Router#
Router# request platform software package install rp 0 file
bootflash:Active_Dir/asr1000rp2-esp*03.13.00.S.154-3.S-ext*.pkg slot 0
--- Starting local lock acquisition on R0 ---
Finished local lock acquisition on R0

--- Starting installation state synchronization ---
Finished installation state synchronization

--- Starting file path checking ---
Finished file path checking

--- Starting image file verification ---
Checking image file names
Locating image files and validating name syntax
  Found asr1000rp2-espbase.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-espx86base.03.13.00.S.154-3.S-ext.pkg
Verifying image file locations
Inspecting image file types
Processing image file constraints
Creating candidate provisioning file

*Aug  4 19:31:14.730 IST: %CPPHA-7-START: F1: cpp_ha:  CPP 0 running init image
/tmp/sw/fp/1/0/fpx86/mount/usr/cpp/bin/qfp-ucode-esp40
*Aug  4 19:31:15.079 IST: %CPPHA-7-READY: F1: cpp_ha:  CPP 0 loading and initialization
completeFinished image file verification

--- Starting candidate package set construction ---
Verifying existing software set

*Aug  4 19:31:15.309 IST: %IOSXE-6-PLATFORM: F1: cpp_cp: Process
CPP_PFILTER_EA_EVENT__API_CALL__REGISTERProcessing candidate provisioning file

*Aug  4 19:31:18.010 IST: %CMRP-6-FP_HA_STATUS: R0/0: cmand:  F0 redundancy state is
Active with ready StandbyConstructing working set for candidate package set
Constructing working set for running package set
Checking command output
Constructing merge of running and candidate packages
Checking if resulting candidate package set would be complete
Finished candidate package set construction

--- Starting compatibility testing ---
Determining whether candidate package set is compatible

WARNING:
WARNING: Candidate software combination not found in compatibility database
WARNING:

Determining whether installation is valid
Creating matrix_file by locate_latest_matrix_file  /tmp/issu/provision/sw
Software sets are identified as compatible
Verifying image type compatibility
Checking IPC compatibility with running software
Checking candidate package set infrastructure compatibility
Checking infrastructure compatibility with running software
Checking package specific compatibility
Finished compatibility testing

--- Starting impact testing ---
Checking operational impact of change
Finished impact testing
```

```
--- Starting list of software package changes ---
Old files list:
  Removed asr1000rp2-espbase.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-espx86base.03.12.01.S.154-2.S.pkg
No new package files added
Finished list of software package changes

--- Starting commit of software changes ---
Updating provisioning rollback files
Creating pending provisioning file
Committing provisioning file
Finished commit of software changes

--- Starting analysis of software changes ---
Finished analysis of software changes

--- Starting update running software ---
Blocking peer synchronization of operating information
Creating the command set placeholder directory
  Finding latest command set
  Finding latest command shortlist lookup file
  Finding latest command shortlist file
  Assembling CLI output libraries
  Assembling CLI input libraries
  Assembling Dynamic configuration files
  Applying interim IPC and database definitions
  Replacing running software
  Replacing CLI software
  Restarting software
    Restarting ESP0
  Applying final IPC and database definitions

*Aug  4 19:32:46.187 IST: %IOSXE_OIR-6-OFFLINECARD: Card (fp) offline in slot F0
*Aug  4 19:32:46.539 IST: %CMRP-6-FP_HA_STATUS: R0/0: cmand:  F1 redundancy state is
Active  Generating software version information
  Notifying running software of updates
  Unblocking peer synchronization of operating information
Unmounting old packages
Cleaning temporary installation files
  Finished update running software

SUCCESS: Finished installing software.

Router#
*Aug  4 19:34:19.748 IST: %CPPHA-7-START: F0: cpp_ha:  CPP 0 preparing image
/tmp/sw/fp/0/0/fpx86/mount/usr/cpp/bin/qfp-ucode-esp40
*Aug  4 19:34:20.139 IST: %CPPHA-7-START: F0: cpp_ha:  CPP 0 startup init image
/tmp/sw/fp/0/0/fpx86/mount/usr/cpp/bin/qfp-ucode-esp40
*Aug  4 19:34:21.858 IST: %IOSXE_OIR-6-ONLINECARD: Card (fp) online in slot F0
*Aug  4 19:34:43.609 IST: %CPPHA-7-START: F0: cpp_ha:  CPP 0 running init image
/tmp/sw/fp/0/0/fpx86/mount/usr/cpp/bin/qfp-ucode-esp40
*Aug  4 19:34:43.958 IST: %CPPHA-7-READY: F0: cpp_ha:  CPP 0 loading and initialization
complete
*Aug  4 19:34:44.190 IST: %IOSXE-6-PLATFORM: F0: cpp_cp: Process
CPP_PFILTER_EA_EVENT__API_CALL__REGISTER
*Aug  4 19:34:46.890 IST: %CMRP-6-FP_HA_STATUS: R0/0: cmand:  F0 redundancy state is
Standby
Router#
Router# show platform
Chassis type: ASR1013

Slot      Type                State                 Insert time (ago)
--------- ------------------- --------------------- -----------------
```

```
 2        ASR1000-SIP40       ok                    1d04h
 2/0      SPA-1X10GE-L-V2     ok                    1d04h
 2/1      SPA-1X10GE-L-V2     ok                    1d04h
 2/2      SPA-1X10GE-L-V2     ok                    1d04h
 2/3      SPA-1X10GE-L-V2     ok                    1d04h
 4        ASR1000-2T+20X1GE   ok                    1d04h
 4/0      BUILT-IN-2T+20X1GE  ok                    1d04h
 R0       ASR1000-RP2         ok, active            1d04h
 R1       ASR1000-RP2         ok, standby           1d04h
 F0       ASR1000-ESP100      ok, standby           1d04h
 F1       ASR1000-ESP100      ok, active            1d04h
 P0       ASR1013-PWR-AC      ok                    1d04h
 P1       ASR1013-PWR-AC      ok                    1d04h
 P2       ASR1013-PWR-AC      ok                    1d04h
 P3       ASR1013-PWR-AC      ps, fail              1d04h

Slot      CPLD Version        Firmware Version
--------- ------------------- ----------------------------------------
2         00200800            15.3(3r)S
4         00200800            15.3(3r)S
R0        10021901            15.3(3r)S
R1        10021901            15.3(3r)S
F0        12071700            15.3(3r)S
F1        12071700            15.3(3r)S

Router#
Router# issu commitversion
--- Starting local lock acquisition on R0 ---
Finished local lock acquisition on R0

--- Starting installation changes ---
Cancelling rollback timer
Finished installation changes

SUCCESS: Installation changes committed
Router#
Router#
Router# request platform software package install rp 0 file
bootflash:Active_Dir/asr1000rp2*03.13.00.S.154-3.S-ext*.pkg force
--- Starting local lock acquisition on R0 ---
Finished local lock acquisition on R0

--- Starting installation state synchronization ---
Finished installation state synchronization

--- Starting file path checking ---
Finished file path checking

--- Starting image file verification ---
Checking image file names
Locating image files and validating name syntax
  Found asr1000rp2-elcbase.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-espbase.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-espx86base.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-rpaccess.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-rpbase.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-rpcontrol.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-rpios-adventerprisek9.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg
  Found asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg
Verifying image file locations
Inspecting image file types
    WARNING: In-service installation of IOSD package
```

```
    WARNING: requires software redundancy on target RP
    WARNING: or on-reboot parameter
    WARNING: Automatically setting the on-reboot flag
    WARNING: In-service installation of RP Base package
    WARNING: requires software reboot of target RP
Processing image file constraints
Creating candidate provisioning file
Finished image file verification

--- Starting candidate package set construction ---
Verifying existing software set
Processing candidate provisioning file
Constructing working set for candidate package set
Constructing working set for running package set
Checking command output
Constructing merge of running and candidate packages
Checking if resulting candidate package set would be complete
Finished candidate package set construction

--- Starting compatibility testing ---
Determining whether candidate package set is compatible
Determining whether installation is valid
Determining whether installation is valid ... skipped
Verifying image type compatibility
Checking IPC compatibility for candidate software
Checking candidate package set infrastructure compatibility
Checking infrastructure compatibility with running software
Checking infrastructure compatibility with running software ... skipped
Checking package specific compatibility
Finished compatibility testing

--- Starting list of software package changes ---
Old files list:
  Removed asr1000rp2-elcbase.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-elcspa.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-rpaccess.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-rpbase.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-rpcontrol.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-rpios-adventerprisek9.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-sipbase.03.12.01.S.154-2.S.pkg
  Removed asr1000rp2-sipspa.03.12.01.S.154-2.S.pkg
New files list:
  Added asr1000rp2-rpaccess.03.13.00.S.154-3.S-ext.pkg
  Added asr1000rp2-rpbase.03.13.00.S.154-3.S-ext.pkg
  Added asr1000rp2-rpcontrol.03.13.00.S.154-3.S-ext.pkg
  Added asr1000rp2-rpios-adventerprisek9.03.13.00.S.154-3.S-ext.pkg
Finished list of software package changes

--- Starting commit of software changes ---
Updating provisioning rollback files
Creating pending provisioning file
Committing provisioning file
Finished commit of software changes

SUCCESS: Software provisioned.  New software will load on reboot.
Router#
Router#
Router# show version R0 provisioned
Package: Provisioning File, version: n/a, status: active
  File: bootflash:Active_Dir/packages.conf, on: RP0
  Built: n/a, by: n/a
  File SHA1 checksum: c79075780592aec1312725f4a2357a034fda2d3b

Package: rpbase, version: 03.13.00.S.154-3.S-ext, status: n/a
```

```
  File: bootflash:Active_Dir/asr1000rp2-rpbase.03.13.00.S.154-3.S-ext.pkg, on: RP0
  Built: 2013-07-25_22.55, by: mcpre
  File SHA1 checksum: 4f655c54bb95b4dfa24a0d25ebf97cf8527c69e9

Package: rpcontrol, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-rpcontrol.03.13.00.S.154-3.S-ext.pkg, on: RP0/0
  Built: 2013-07-25_22.55, by: mcpre
  File SHA1 checksum: 8a0a45ea5c7a656c0eef6726174461584f182c78

Package: rpios-adventerprisek9, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-rpios-adventerprisek9.03.13.00.S.154-3.S-ext.pkg,
on: RP0/0
  Built: 2013-07-25_23.00, by: mcpre
  File SHA1 checksum: 85e9eab826bff2194ef568a56c76453625383ad2

Package: rpaccess, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-rpaccess.03.13.00.S.154-3.S-ext.pkg, on: RP0/0
  Built: 2013-07-25_22.55, by: mcpre
  File SHA1 checksum: a360dff0fd76a9b1ae67cda9116c97b62f25ab09

Package: rpcontrol, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-rpcontrol.03.13.00.S.154-3.S-ext.pkg, on: RP0/1
  Built: 2013-07-25_22.55, by: mcpre
  File SHA1 checksum: 8a0a45ea5c7a656c0eef6726174461584f182c78

Package: rpios-adventerprisek9, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-rpios-adventerprisek9.03.13.00.S.154-3.S-ext.pkg,
on: RP0/1
  Built: 2013-07-25_23.00, by: mcpre
  File SHA1 checksum: 85e9eab826bff2194ef568a56c76453625383ad2

Package: rpaccess, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-rpaccess.03.13.00.S.154-3.S-ext.pkg, on: RP0/1
  Built: 2013-07-25_22.55, by: mcpre
  File SHA1 checksum: a360dff0fd76a9b1ae67cda9116c97b62f25ab09

Package: rpbase, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-rpbase.03.13.00.S.154-3.S-ext.pkg, on: RP1
  Built: 2013-07-25_22.55, by: mcpre
  File SHA1 checksum: 4f655c54bb95b4dfa24a0d25ebf97cf8527c69e9

Package: rpcontrol, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-rpcontrol.03.13.00.S.154-3.S-ext.pkg, on: RP1/0
  Built: 2013-07-25_22.55, by: mcpre
  File SHA1 checksum: 8a0a45ea5c7a656c0eef6726174461584f182c78

Package: rpios-adventerprisek9, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-rpios-adventerprisek9.03.13.00.S.154-3.S-ext.pkg,
on: RP1/0
  Built: 2013-07-25_23.00, by: mcpre
  File SHA1 checksum: 85e9eab826bff2194ef568a56c76453625383ad2

Package: rpaccess, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-rpaccess.03.13.00.S.154-3.S-ext.pkg, on: RP1/0
  Built: 2013-07-25_22.55, by: mcpre
  File SHA1 checksum: a360dff0fd76a9b1ae67cda9116c97b62f25ab09

Package: rpcontrol, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-rpcontrol.03.13.00.S.154-3.S-ext.pkg, on: RP1/1
  Built: 2013-07-25_22.55, by: mcpre
  File SHA1 checksum: 8a0a45ea5c7a656c0eef6726174461584f182c78

Package: rpios-adventerprisek9, version: 03.13.00.S.154-3.S-ext, status: n/a
```

```
  File: bootflash:Active_Dir/asr1000rp2-rpios-adventerprisek9.03.13.00.S.154-3.S-ext.pkg,
on: RP1/1
  Built: 2013-07-25_23.00, by: mcpre
  File SHA1 checksum: 85e9eab826bff2194ef568a56c76453625383ad2

Package: rpaccess, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-rpaccess.03.13.00.S.154-3.S-ext.pkg, on: RP1/1
  Built: 2013-07-25_22.55, by: mcpre
  File SHA1 checksum: a360dff0fd76a9b1ae67cda9116c97b62f25ab09

Package: espbase, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-espbase.03.13.00.S.154-3.S-ext.pkg, on: ESP0
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 2fe0ede1545e3f8260b7d453653e812500f0d7b0

Package: espx86base, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-espx86base.03.13.00.S.154-3.S-ext.pkg, on: ESP0
  Built: 2013-07-25_22.55, by: mcpre
  File SHA1 checksum: 571b8bb3866341badd6e24de677b98409f0c789c

Package: espbase, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-espbase.03.13.00.S.154-3.S-ext.pkg, on: ESP1
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 2fe0ede1545e3f8260b7d453653e812500f0d7b0

Package: espx86base, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-espx86base.03.13.00.S.154-3.S-ext.pkg, on: ESP1
  Built: 2013-07-25_22.55, by: mcpre
  File SHA1 checksum: 571b8bb3866341badd6e24de677b98409f0c789c

Package: sipbase, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg, on: SIP0
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 3b6a4838972840a995ff22e73fd2bae910b268a7

Package: elcbase, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcbase.03.13.00.S.154-3.S-ext.pkg, on: SIP0
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 99f8dc925083b118626a4e82d93079050db96826

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP0/0
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP0/0
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP0/1
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP0/1
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP0/2
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca
```

```
Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP0/2
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP0/3
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP0/3
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipbase, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg, on: SIP1
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 3b6a4838972840a995ff22e73fd2bae910b268a7

Package: elcbase, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcbase.03.13.00.S.154-3.S-ext.pkg, on: SIP1
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 99f8dc925083b118626a4e82d93079050db96826

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP1/0
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP1/0
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP1/1
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP1/1
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP1/2
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP1/2
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP1/3
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP1/3
  Built: 2013-07-25_21.16, by: mcpre
```

```
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipbase, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg, on: SIP2
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 3b6a4838972840a995ff22e73fd2bae910b268a7

Package: elcbase, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcbase.03.13.00.S.154-3.S-ext.pkg, on: SIP2
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 99f8dc925083b118626a4e82d93079050db96826

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP2/0
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP2/0
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP2/1
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP2/1
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP2/2
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP2/2
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP2/3
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP2/3
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipbase, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg, on: SIP3
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 3b6a4838972840a995ff22e73fd2bae910b268a7

Package: elcbase, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcbase.03.13.00.S.154-3.S-ext.pkg, on: SIP3
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 99f8dc925083b118626a4e82d93079050db96826

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP3/0
```

```
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP3/0
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP3/1
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP3/1
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP3/2
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP3/2
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP3/3
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP3/3
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipbase, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg, on: SIP4
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 3b6a4838972840a995ff22e73fd2bae910b268a7

Package: elcbase, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcbase.03.13.00.S.154-3.S-ext.pkg, on: SIP4
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 99f8dc925083b118626a4e82d93079050db96826

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP4/0
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP4/0
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP4/1
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
```

```
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP4/1
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP4/2
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP4/2
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP4/3
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP4/3
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipbase, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg, on: SIP5
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 3b6a4838972840a995ff22e73fd2bae910b268a7

Package: elcbase, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcbase.03.13.00.S.154-3.S-ext.pkg, on: SIP5
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 99f8dc925083b118626a4e82d93079050db96826

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP5/0
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP5/0
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP5/1
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP5/1
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP5/2
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP5/2
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499
```

```
Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP5/3
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP5/3
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499


Router#
Router#
Router# show version R0 provisioned
Package: Provisioning File, version: n/a, status: active
  File: bootflash:Active_Dir/packages.conf, on: RP0
  Built: n/a, by: n/a
  File SHA1 checksum: c79075780592aec1312725f4a2357a034fda2d3b

Package: rpbase, version: 03.12.01.S.154-2.S, status: active
  File: bootflash:Active_Dir/asr1000rp2-rpbase.03.12.01.S.154-2.S.pkg, on: RP0
  Built: 2013-03-25_18.48, by: mcpre
  File SHA1 checksum: 3a9675142898cfac350d4e42f0e37bd9f4e48538

Package: rpcontrol, version: 03.12.01.S.154-2.S, status: active
  File: bootflash:Active_Dir/asr1000rp2-rpcontrol.03.12.01.S.154-2.S.pkg, on: RP0/0
  Built: 2013-03-25_18.48, by: mcpre
  File SHA1 checksum: 87b11f863f67fdf2610ee0769b929baab4c3efad

Package: rpios-adventerprisek9, version: 03.12.01.S.154-2.S, status: active
  File: bootflash:Active_Dir/asr1000rp2-rpios-adventerprisek9.03.12.01.S.154-2.S.pkg, on:
RP0/0
  Built: 2013-03-25_18.51, by: mcpre
  File SHA1 checksum: b487136319da0a327844d353c77e533c53c56053

Package: rpaccess, version: 03.12.01.S.154-2.S, status: active
  File: bootflash:Active_Dir/asr1000rp2-rpaccess.03.12.01.S.154-2.S.pkg, on: RP0/0
  Built: 2013-03-25_18.48, by: mcpre
  File SHA1 checksum: 032bea36f74b19977b363243c99f02413b54104d

Package: rpcontrol, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-rpcontrol.03.12.01.S.154-2.S.pkg, on: RP0/1
  Built: 2013-03-25_18.48, by: mcpre
  File SHA1 checksum: 87b11f863f67fdf2610ee0769b929baab4c3efad

Package: rpios-adventerprisek9, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-rpios-adventerprisek9.03.12.01.S.154-2.S.pkg, on:
RP0/1
  Built: 2013-03-25_18.51, by: mcpre
  File SHA1 checksum: b487136319da0a327844d353c77e533c53c56053

Package: rpaccess, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-rpaccess.03.12.01.S.154-2.S.pkg, on: RP0/1
  Built: 2013-03-25_18.48, by: mcpre
  File SHA1 checksum: 032bea36f74b19977b363243c99f02413b54104d

Package: rpbase, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-rpbase.03.12.01.S.154-2.S.pkg, on: RP1
  Built: 2013-03-25_18.48, by: mcpre
  File SHA1 checksum: 3a9675142898cfac350d4e42f0e37bd9f4e48538

Package: rpcontrol, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-rpcontrol.03.12.01.S.154-2.S.pkg, on: RP1/0
  Built: 2013-03-25_18.48, by: mcpre
```

```
  File SHA1 checksum: 87b11f863f67fdf2610ee0769b929baab4c3efad

Package: rpios-adventerprisek9, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-rpios-adventerprisek9.03.12.01.S.154-2.S.pkg, on:
RP1/0
  Built: 2013-03-25_18.51, by: mcpre
  File SHA1 checksum: b487136319da0a327844d353c77e533c53c56053

Package: rpaccess, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-rpaccess.03.12.01.S.154-2.S.pkg, on: RP1/0
  Built: 2013-03-25_18.48, by: mcpre
  File SHA1 checksum: 032bea36f74b19977b363243c99f02413b54104d

Package: rpcontrol, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-rpcontrol.03.12.01.S.154-2.S.pkg, on: RP1/1
  Built: 2013-03-25_18.48, by: mcpre
  File SHA1 checksum: 87b11f863f67fdf2610ee0769b929baab4c3efad

Package: rpios-adventerprisek9, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-rpios-adventerprisek9.03.12.01.S.154-2.S.pkg, on:
RP1/1
  Built: 2013-03-25_18.51, by: mcpre
  File SHA1 checksum: b487136319da0a327844d353c77e533c53c56053

Package: rpaccess, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-rpaccess.03.12.01.S.154-2.S.pkg, on: RP1/1
  Built: 2013-03-25_18.48, by: mcpre
  File SHA1 checksum: 032bea36f74b19977b363243c99f02413b54104d

Package: espbase, version: 03.13.00.S.154-3.S-ext, status: inactive
  File: bootflash:Active_Dir/asr1000rp2-espbase.03.13.00.S.154-3.S-ext.pkg, on: ESP0
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 2fe0ede1545e3f8260b7d453653e812500f0d7b0

Package: espx86base, version: 03.13.00.S.154-3.S-ext, status: active
  File: bootflash:Active_Dir/asr1000rp2-espx86base.03.13.00.S.154-3.S-ext.pkg, on: ESP0
  Built: 2013-07-25_22.55, by: mcpre
  File SHA1 checksum: 571b8bb3866341badd6e24de677b98409f0c789c

Package: espbase, version: 03.13.00.S.154-3.S-ext, status: inactive
  File: bootflash:Active_Dir/asr1000rp2-espbase.03.13.00.S.154-3.S-ext.pkg, on: ESP1
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 2fe0ede1545e3f8260b7d453653e812500f0d7b0

Package: espx86base, version: 03.13.00.S.154-3.S-ext, status: active
  File: bootflash:Active_Dir/asr1000rp2-espx86base.03.13.00.S.154-3.S-ext.pkg, on: ESP1
  Built: 2013-07-25_22.55, by: mcpre
  File SHA1 checksum: 571b8bb3866341badd6e24de677b98409f0c789c

Package: sipbase, version: 03.12.01.S.154-2.S, status: inactive
  File: bootflash:Active_Dir/asr1000rp2-sipbase.03.12.01.S.154-2.S.pkg, on: SIP0
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: fb815b5cbaf5fd20a0a0e2aeabd2687347c6921d

Package: elcbase, version: 03.12.01.S.154-2.S, status: inactive
  File: bootflash:Active_Dir/asr1000rp2-elcbase.03.12.01.S.154-2.S.pkg, on: SIP0
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: fb1d6abd055b191909bc78ccac23b964de15ab8e

Package: sipspa, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.12.01.S.154-2.S.pkg, on: SIP0/0
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 644364aeea8ccebdd4af5b8d29367db50fc82b17
```

```
Package: elcspa, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.12.01.S.154-2.S.pkg, on: SIP0/0
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 2e6b6b1949261873ce5ce189ec19440abffd71c6

Package: sipspa, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.12.01.S.154-2.S.pkg, on: SIP0/1
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 644364aeea8ccebdd4af5b8d29367db50fc82b17

Package: elcspa, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.12.01.S.154-2.S.pkg, on: SIP0/1
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 2e6b6b1949261873ce5ce189ec19440abffd71c6

Package: sipspa, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.12.01.S.154-2.S.pkg, on: SIP0/2
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 644364aeea8ccebdd4af5b8d29367db50fc82b17

Package: elcspa, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.12.01.S.154-2.S.pkg, on: SIP0/2
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 2e6b6b1949261873ce5ce189ec19440abffd71c6

Package: sipspa, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.12.01.S.154-2.S.pkg, on: SIP0/3
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 644364aeea8ccebdd4af5b8d29367db50fc82b17

Package: elcspa, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.12.01.S.154-2.S.pkg, on: SIP0/3
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 2e6b6b1949261873ce5ce189ec19440abffd71c6

Package: sipbase, version: 03.12.01.S.154-2.S, status: inactive
  File: bootflash:Active_Dir/asr1000rp2-sipbase.03.12.01.S.154-2.S.pkg, on: SIP1
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: fb815b5cbaf5fd20a0a0e2aeabd2687347c6921d

Package: elcbase, version: 03.12.01.S.154-2.S, status: inactive
  File: bootflash:Active_Dir/asr1000rp2-elcbase.03.12.01.S.154-2.S.pkg, on: SIP1
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: fb1d6abd055b191909bc78ccac23b964de15ab8e

Package: sipspa, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.12.01.S.154-2.S.pkg, on: SIP1/0
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 644364aeea8ccebdd4af5b8d29367db50fc82b17

Package: elcspa, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.12.01.S.154-2.S.pkg, on: SIP1/0
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 2e6b6b1949261873ce5ce189ec19440abffd71c6

Package: sipspa, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.12.01.S.154-2.S.pkg, on: SIP1/1
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 644364aeea8ccebdd4af5b8d29367db50fc82b17

Package: elcspa, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.12.01.S.154-2.S.pkg, on: SIP1/1
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 2e6b6b1949261873ce5ce189ec19440abffd71c6
```

```
Package: sipspa, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.12.01.S.154-2.S.pkg, on: SIP1/2
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 644364aeea8ccebdd4af5b8d29367db50fc82b17

Package: elcspa, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.12.01.S.154-2.S.pkg, on: SIP1/2
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 2e6b6b1949261873ce5ce189ec19440abffd71c6

Package: sipspa, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.12.01.S.154-2.S.pkg, on: SIP1/3
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 644364aeea8ccebdd4af5b8d29367db50fc82b17

Package: elcspa, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.12.01.S.154-2.S.pkg, on: SIP1/3
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 2e6b6b1949261873ce5ce189ec19440abffd71c6

Package: sipbase, version: 03.13.00.S.154-3.S-ext, status: active
  File: bootflash:Active_Dir/asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg, on: SIP2
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 3b6a4838972840a995ff22e73fd2bae910b268a7

Package: elcbase, version: 03.12.01.S.154-2.S, status: inactive
  File: bootflash:Active_Dir/asr1000rp2-elcbase.03.12.01.S.154-2.S.pkg, on: SIP2
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: fb1d6abd055b191909bc78ccac23b964de15ab8e

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: active
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP2/0
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.12.01.S.154-2.S, status: inactive
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.12.01.S.154-2.S.pkg, on: SIP2/0
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 2e6b6b1949261873ce5ce189ec19440abffd71c6

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: active
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP2/1
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.12.01.S.154-2.S, status: inactive
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.12.01.S.154-2.S.pkg, on: SIP2/1
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 2e6b6b1949261873ce5ce189ec19440abffd71c6

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: active
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP2/2
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.12.01.S.154-2.S, status: inactive
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.12.01.S.154-2.S.pkg, on: SIP2/2
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 2e6b6b1949261873ce5ce189ec19440abffd71c6

Package: sipspa, version: 03.13.00.S.154-3.S-ext, status: active
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg, on: SIP2/3
  Built: 2013-07-25_21.16, by: mcpre
```

```
  File SHA1 checksum: 6d12280b5cc33d17d752f475bf340b77ef3451ca

Package: elcspa, version: 03.12.01.S.154-2.S, status: inactive
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.12.01.S.154-2.S.pkg, on: SIP2/3
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 2e6b6b1949261873ce5ce189ec19440abffd71c6

Package: sipbase, version: 03.12.01.S.154-2.S, status: inactive
  File: bootflash:Active_Dir/asr1000rp2-sipbase.03.12.01.S.154-2.S.pkg, on: SIP3
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: fb815b5cbaf5fd20a0a0e2aeabd2687347c6921d

Package: elcbase, version: 03.12.01.S.154-2.S, status: inactive
  File: bootflash:Active_Dir/asr1000rp2-elcbase.03.12.01.S.154-2.S.pkg, on: SIP3
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: fb1d6abd055b191909bc78ccac23b964de15ab8e

Package: sipspa, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.12.01.S.154-2.S.pkg, on: SIP3/0
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 644364aeea8ccebdd4af5b8d29367db50fc82b17

Package: elcspa, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.12.01.S.154-2.S.pkg, on: SIP3/0
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 2e6b6b1949261873ce5ce189ec19440abffd71c6

Package: sipspa, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.12.01.S.154-2.S.pkg, on: SIP3/1
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 644364aeea8ccebdd4af5b8d29367db50fc82b17

Package: elcspa, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.12.01.S.154-2.S.pkg, on: SIP3/1
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 2e6b6b1949261873ce5ce189ec19440abffd71c6

Package: sipspa, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.12.01.S.154-2.S.pkg, on: SIP3/2
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 644364aeea8ccebdd4af5b8d29367db50fc82b17

Package: elcspa, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.12.01.S.154-2.S.pkg, on: SIP3/2
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 2e6b6b1949261873ce5ce189ec19440abffd71c6

Package: sipspa, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.12.01.S.154-2.S.pkg, on: SIP3/3
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 644364aeea8ccebdd4af5b8d29367db50fc82b17

Package: elcspa, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.12.01.S.154-2.S.pkg, on: SIP3/3
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 2e6b6b1949261873ce5ce189ec19440abffd71c6

Package: sipbase, version: 03.12.01.S.154-2.S, status: inactive
  File: bootflash:Active_Dir/asr1000rp2-sipbase.03.12.01.S.154-2.S.pkg, on: SIP4
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: fb815b5cbaf5fd20a0a0e2aeabd2687347c6921d

Package: elcbase, version: 03.13.00.S.154-3.S-ext, status: active
  File: bootflash:Active_Dir/asr1000rp2-elcbase.03.13.00.S.154-3.S-ext.pkg, on: SIP4
```

```
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 99f8dc925083b118626a4e82d93079050db96826

Package: sipspa, version: 03.12.01.S.154-2.S, status: inactive
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.12.01.S.154-2.S.pkg, on: SIP4/0
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 644364aeea8ccebdd4af5b8d29367db50fc82b17

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: active
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP4/0
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipspa, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.12.01.S.154-2.S.pkg, on: SIP4/1
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 644364aeea8ccebdd4af5b8d29367db50fc82b17

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP4/1
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipspa, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.12.01.S.154-2.S.pkg, on: SIP4/2
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 644364aeea8ccebdd4af5b8d29367db50fc82b17

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP4/2
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipspa, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.12.01.S.154-2.S.pkg, on: SIP4/3
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 644364aeea8ccebdd4af5b8d29367db50fc82b17

Package: elcspa, version: 03.13.00.S.154-3.S-ext, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg, on: SIP4/3
  Built: 2013-07-25_21.16, by: mcpre
  File SHA1 checksum: 94763274fc807489410e299a45fd73fce9d67499

Package: sipbase, version: 03.12.01.S.154-2.S, status: inactive
  File: bootflash:Active_Dir/asr1000rp2-sipbase.03.12.01.S.154-2.S.pkg, on: SIP5
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: fb815b5cbaf5fd20a0a0e2aeabd2687347c6921d

Package: elcbase, version: 03.12.01.S.154-2.S, status: inactive
  File: bootflash:Active_Dir/asr1000rp2-elcbase.03.12.01.S.154-2.S.pkg, on: SIP5
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: fb1d6abd055b191909bc78ccac23b964de15ab8e

Package: sipspa, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.12.01.S.154-2.S.pkg, on: SIP5/0
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 644364aeea8ccebdd4af5b8d29367db50fc82b17

Package: elcspa, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.12.01.S.154-2.S.pkg, on: SIP5/0
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 2e6b6b1949261873ce5ce189ec19440abffd71c6

Package: sipspa, version: 03.12.01.S.154-2.S, status: n/a
```

```
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.12.01.S.154-2.S.pkg, on: SIP5/1
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 644364aeea8ccebdd4af5b8d29367db50fc82b17

Package: elcspa, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.12.01.S.154-2.S.pkg, on: SIP5/1
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 2e6b6b1949261873ce5ce189ec19440abffd71c6

Package: sipspa, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.12.01.S.154-2.S.pkg, on: SIP5/2
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 644364aeea8ccebdd4af5b8d29367db50fc82b17

Package: elcspa, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.12.01.S.154-2.S.pkg, on: SIP5/2
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 2e6b6b1949261873ce5ce189ec19440abffd71c6

Package: sipspa, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-sipspa.03.12.01.S.154-2.S.pkg, on: SIP5/3
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 644364aeea8ccebdd4af5b8d29367db50fc82b17

Package: elcspa, version: 03.12.01.S.154-2.S, status: n/a
  File: bootflash:Active_Dir/asr1000rp2-elcspa.03.12.01.S.154-2.S.pkg, on: SIP5/3
  Built: 2013-03-25_17.28, by: mcpre
  File SHA1 checksum: 2e6b6b1949261873ce5ce189ec19440abffd71c6


Router#
Router#redundancy force-switchover
Proceed with switchover to standby RP? [confirm]
<output removed for brevity>
Router#
Router# request platform software package clean
Cleaning up unnecessary package files
No path specified, will use booted path bootflash:Active_Dir/packages.conf
Cleaning bootflash:Active_Dir
  Scanning boot directory for packages ... done.
  Preparing packages list to delete ...
    asr1000rp2-elcbase.03.13.00.S.154-3.S-ext.pkg
      File is in use, will not delete.
    asr1000rp2-elcspa.03.13.00.S.154-3.S-ext.pkg
      File is in use, will not delete.
    asr1000rp2-espbase.03.13.00.S.154-3.S-ext.pkg
      File is in use, will not delete.
    asr1000rp2-espx86base.03.13.00.S.154-3.S-ext.pkg
      File is in use, will not delete.
    asr1000rp2-rpaccess.03.13.00.S.154-3.S-ext.pkg
      File is in use, will not delete.
    asr1000rp2-rpbase.03.13.00.S.154-3.S-ext.pkg
      File is in use, will not delete.
    asr1000rp2-rpcontrol.03.13.00.S.154-3.S-ext.pkg
      File is in use, will not delete.
    asr1000rp2-rpios-adventerprisek9.03.13.00.S.154-3.S-ext.pkg
      File is in use, will not delete.
    asr1000rp2-sipbase.03.13.00.S.154-3.S-ext.pkg
      File is in use, will not delete.
    asr1000rp2-sipspa.03.13.00.S.154-3.S-ext.pkg
      File is in use, will not delete.
    packages.conf
      File is in use, will not delete.
  done.
```

```
Files that will be deleted:
  asr1000rp2-elcbase.03.12.01.S.154-2.S.pkg
  asr1000rp2-elcspa.03.12.01.S.154-2.S.pkg
  asr1000rp2-espbase.03.12.01.S.154-2.S.pkg
  asr1000rp2-espx86base.03.12.01.S.154-2.S.pkg
  asr1000rp2-packages-adventerprisek9.03.12.01.S.154-2.S.conf
  asr1000rp2-rpaccess.03.12.01.S.154-2.S.pkg
  asr1000rp2-rpbase.03.12.01.S.154-2.S.pkg
  asr1000rp2-rpcontrol.03.12.01.S.154-2.S.pkg
  asr1000rp2-rpios-adventerprisek9.03.12.01.S.154-2.S.pkg
  asr1000rp2-sipbase.03.12.01.S.154-2.S.pkg
  asr1000rp2-sipspa.03.12.01.S.154-2.S.pkg
  packages.conf.00-

Do you want to proceed? [confirm]y
  Deleting file bootflash:Active_Dir/asr1000rp2-elcbase.03.12.01.S.154-2.S.pkg ... done.
  Deleting file bootflash:Active_Dir/asr1000rp2-elcspa.03.12.01.S.154-2.S.pkg ... done.
  Deleting file bootflash:Active_Dir/asr1000rp2-espbase.03.12.01.S.154-2.S.pkg ... done.
  Deleting file bootflash:Active_Dir/asr1000rp2-espx86base.03.12.01.S.154-2.S.pkg ...
done.
  Deleting file
bootflash:Active_Dir/asr1000rp2-packages-adventerprisek9.03.12.01.S.154-2.S.conf ... done.
  Deleting file bootflash:Active_Dir/asr1000rp2-rpaccess.03.12.01.S.154-2.S.pkg ... done.
  Deleting file bootflash:Active_Dir/asr1000rp2-rpbase.03.12.01.S.154-2.S.pkg ... done.
  Deleting file bootflash:Active_Dir/asr1000rp2-rpcontrol.03.12.01.S.154-2.S.pkg ... done.
  Deleting file
bootflash:Active_Dir/asr1000rp2-rpios-adventerprisek9.03.12.01.S.154-2.S.pkg ... done.
  Deleting file bootflash:Active_Dir/asr1000rp2-sipbase.03.12.01.S.154-2.S.pkg ... done.
  Deleting file bootflash:Active_Dir/asr1000rp2-sipspa.03.12.01.S.154-2.S.pkg ... done.
  Deleting file bootflash:Active_Dir/packages.conf.00- ... done.
SUCCESS: Files deleted.
Router#
Router#
```

Additional References

The following sections provide references related to the Software Upgrade Process feature.

Related Documents

Related Topic	Title
Cisco IOS XE commands	*Cisco IOS Master Commands List, All Releases*
Performing an In Service Software Upgrade	*High Availability Configuration Guide, Cisco IOS XE Release 3S*

Standards

Standard	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

MIBs

MIB	MIBs Link
None	To locate and download MIBs for selected platforms, Cisco IOS XE software releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

RFC	Title
No new or modified RFCs are supported by this feature, and support for existing RFCs has not been modified by this feature.	—

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for Software Upgrade Process

Table 5-3 lists the release history for this feature on the Cisco ASR 1000 Series Routers.

For information on a feature in this technology that is not documented here, see the *Cisco ASR 1000 Series Aggregation Services Routers Documentation Roadmap*.

Use Cisco Feature Navigator to find information about platform support and software image support. Cisco Feature Navigator enables you to determine which Cisco IOS XE software images support a specific software release, feature set, or platform. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Note Table 5-3 lists only the Cisco IOS XE software releases that introduced support for a given feature in a given Cisco IOS XE software release train. Unless noted otherwise, subsequent releases of that Cisco IOS XE software release train also support that feature.

Table 5-3 Feature Information for Software Upgrade Process

Feature Name	Releases	Feature Information
MDR Support for Cisco ASR 1000 Series Fixed Ethernet Line Card	Cisco IOS XE Release 3.12S	This feature was introduced. Added MDR support for the Cisco ASR 1000 Series Fixed Ethernet Line Card (ASR1000-2T+20X1GE).
Cisco ASR1000 Series Fixed Ethernet Line Card	Cisco IOS XE Release 3.10S	This feature was introduced. Added support for the Cisco ASR 1000 Series Fixed Ethernet Line Card.
Minimal Disruptive Restart ISSU	Cisco IOS XE Release 3.8S	This feature was introduced. Added support for the following GigabitEthernet SPAs to the Minimal Disruptive Restart (MDR) feature on Cisco ASR 1000 Series Router with SIP40: -SPA-2X1GE-V2 -SPA-5X1GE-V2 -SPA-8X1GE-V2 -SPA-10X1GE-V2 -SPA-1X10GE-L-V2
Software Upgrade Process	Cisco IOS XE Release 2.6	This feature was introduced. Cisco ASR 1000 Series Routers support the following software upgrade procedures: • In Service Software Upgrades (ISSU) for redundant platforms • Upgrade process with service impact for nonredundant platforms.

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: www.cisco.com/go/trademarks. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1110R)

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.

Cisco ASR 1000 Series Aggregation Services Routers Software Configuration Guide

Copyright © 2008-2014 Cisco Systems, Inc. All rights reserved.

High Availability Overview

Cisco High Availability (HA) enables network-wide protection by providing fast recovery from faults that may occur in any part of the network. With Cisco High Availability, network hardware and software work together and enable rapid recovery from disruptions to ensure fault transparency to users and network applications.

The unique hardware and software architecture of the Cisco ASR 1000 Series Routers is designed to maximize router uptime during any network event, and thereby provide maximum uptime and resilience within any network scenario.

This guide covers the aspects of High Availability that are unique to the Cisco ASR 1000 Series Routers. It is not intended as a comprehensive guide to High Availability, nor is it intended to provide information on High Availability features that are available on other Cisco routers that are configured and implemented identically on the Cisco ASR 1000 Series Routers. The Cisco IOS feature documents and guides should be used in conjunction with this chapter to gather information about High Availability-related features that are available on multiple Cisco platforms and work identically on the Cisco ASR 1000 Series Routers.

Finding Feature Information in This Module

Your software release might not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the "Feature Information for High Availability Overview" section on page 6-10.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Contents

This section discusses various aspects of High Availability on the Cisco ASR 1000 Series Routers and contains the following sections:

- Hardware Redundancy Overview on the Cisco ASR 1000 Series Routers, page 6-2
- Software Redundancy on the Cisco ASR 1000 Series Routers, page 6-4
- Route Processor Redundancy, page 6-6
- Stateful Switchover, page 6-7

- IPsec Failover, page 6-8
- Bidirectional Forwarding Detection, page 6-8

Hardware Redundancy Overview on the Cisco ASR 1000 Series Routers

Some models of the Cisco ASR 1000 Series Routers offer hardware redundancy within the same Cisco ASR 1000 Series Router through the following methods:

- Allowing two Route Processors (RPs) in the same Cisco ASR 1000 Series Router
- Allowing two Enhanced Services Processors (ESPs) in the same Cisco ASR 1000 Series Router

No hardware redundancy is supported for the following hardware:

- SPA interface processors (SIPs)—A SIP must be reloaded, and traffic briefly interrupted, for a SIP upgrade to complete.
- Shared port adapters (SPAs)—A SPA must be reloaded, which will briefly interrupt traffic to that SPA, for a SPA software subpackage update to complete.

Hardware redundancy on the Cisco ASR 1000 Series Routers gives users the following benefits:

- A failover option—If a processor fails, the standby processor immediately becomes the active processor with little or no delay. The failover happens completely within the same router, so a second standby router is not needed.
- No downtime upgrades—Using features like ISSU, a software upgrade can be handled on the standby processor while the active processor continues normal operation.

Hardware redundancy is available on the Cisco ASR 1006 Router only at this time.

provides a hardware redundancy overview.

Table 6-1 Hardware Redundancy Overview

Hardware	Support for Dual Hardware Configuration on Cisco ASR 1001 Router	Support for Dual Hardware Configuration on Cisco ASR 1002 Router	Support for Dual Hardware Configuration on Cisco ASR 1004 Router	Support for Dual Hardware Configuration on Cisco ASR 1006 Router	Failover Behavior
Enhanced Services Processor	No	No	No	Yes	If an active ESP experiences a hardware or software event that makes it unable to forward traffic (such as a hardware failure, an OIR, or a manual switch) and a standby ESP is configured, the standby ESP becomes the active ESP with the possibility of a minor interruption (less than 200 ms).
Route Processor	No	No	No	Yes	If an active RP experiences an event that makes it unable to forward traffic (such as a hardware failure, a software failure, an OIR, or a manual switch) and a standby RP is configured, the standby RP immediately becomes the active RP.
SPA	No	No	No	No	No standby configurations are available for SPAs. If a SPA fails, that particular SPA is down and unable to forward traffic. In the event of a SPA shutdown, all other SIPs and SPAs on the router continue to be fully operational.
SIP	No	No	No	No	No standby configurations are available for SIPs. If a SIP fails, all SPAs in that SIP are down and unable to forward traffic. In the event of a SIP shutdown, all other SIPs and SPAs on the router continue to be fully operational.

Software Redundancy on the Cisco ASR 1000 Series Routers

This section covers the following topics:

- Software Redundancy Overview, page 6-4
- Second IOS Process on a Cisco ASR 1002 or 1004 Router, page 6-5
- SSO-Aware Protocol and Applications, page 6-7

Software Redundancy Overview

On the Cisco ASR 1000 Series Routers, IOS runs as one of many processes within the operating system. This is different than on traditional Cisco IOS, where all processes are run within Cisco IOS. See the "IOS as a Process" section on page 1-7 for more information regarding IOS as a process on the Cisco ASR 1000 Series Router.

This architecture allows for software redundancy opportunities that are not available on other platforms that run Cisco IOS software. Specifically, a standby IOS process can be available on the same Route Processor as the active IOS process. This standby IOS process can be switched to in the event of an IOS failure, and can also be used to upgrade subpackage software in some scenarios as the standby IOS process in an ISSU upgrade.

On the Cisco ASR 1006 Router, the second IOS process can run only on the standby Route Processor. Two IOS processes on the same Router Processor are not possible for any Cisco ASR 1000 Series Router that supports dual RP hardware redundancy configurations since the second Route Processor can support a standby IOS process. An overview of software redundancy is shown in Table 6-2.

Table 6-2 Software Redundancy Overview

Router	Support for Two IOS Processes on Same Route Processor	Support for a Second IOS Process on Standby Route Processor	Explanation
Cisco ASR 1001 Router[1]	Yes	N/A	The Cisco ASR 1001 Router only supports one RP, so dual IOS processes run on the lone RP.
Cisco ASR 1002 Router	Yes	N/A	The Cisco ASR 1002 Router only supports one RP, so dual IOS processes run on the lone RP.
Cisco ASR 1004 Router	Yes	N/A	The Cisco ASR 1004 Router only supports one RP, so dual IOS processes run on the lone RP.
Cisco ASR 1006 Router	No	Yes	The Cisco ASR 1006 Router supports a second Route Processor, so the second IOS process can only run on the standby Route Processor.

1. If a critical process, such as the ESP or the SIP fails on the Cisco ASR 1001 Router, then the entire chassis reloads.

Second IOS Process on a Cisco ASR 1002 or 1004 Router

For Cisco ASR 1002 and 1004 routers, Route Processor Redundancy and Stateful Switchover can be used to switch between IOS processes. RPR and SSO need to be configured by the user, however, because a second IOS process is not available by default on Cisco ASR 1002 and 1004 routers.

Table 6-2 summarizes the software redundancy opportunities available with the second IOS process for the Cisco ASR 1002 and 1004 routers.

Table 6-3 Software Redundancy Options for Cisco ASR 1002 and 1004 Routers

Router	Default HA Setting	Options with 2 GB or DRAM	Options with 4 GB or DRAM
Cisco ASR 1002 Router	None	None	None, RPR, SSO
Cisco ASR 1004 Router	None	None	None, RPR, SSO

ISSU cannot be used to upgrade consolidated packages on Cisco ASR 1002 or 1004 Routers, and only a few subpackages can be upgraded individually using ISSU through the use of dual IOS processes on the same Route Processor. See the "Route Processor Redundancy" section on page 6-6 for more information on which subpackages can be upgraded using ISSU in a dual RP setup.

Configuring two Cisco IOS process on one RP

On the Cisco ASR 1000 Series Routers, Cisco IOS runs as one of the many processes. This architecture supports software redundancy opportunities. Specifically, a standby Cisco IOS process is available on the same Route Processor as the active Cisco IOS process. In the event of a Cisco IOS failure, the system switches to the standby Cisco IOS process. It also supports software upgrade of subpackages when the standby Cisco IOS process is performing an ISSU upgrade.

This section describes how to configure two Cisco IOS process on one RP.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **redundancy**
4. **mode SSO**
5. **exit**
6. **reload**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**redundancy** **Example:** `Router(config)# redundancy`	Enters redundancy configuration mode.
Step 4	**mode SSO** **Example:** `Router(config)# mode SSO`	Configures SSO. When this command is entered, the redundant supervisor engine is reloaded and begins to work in SSO mode.

Step 5	**exit**
	
	Example:
	`Router(config)# exit`
	`Router #`
		Exits configuration mode and returns to global configuration mode.

Step 6	**reload** **Example:** `Router # reload`	Reloads IOS.

Example

```
Router# configure terminal
Router(config)# redundancy
Router(config)# mode SSO
Router(config)# exit
Router# reload
```

Route Processor Redundancy

Route Processor Redundancy (RPR) allows you to configure a standby RP. When you configure RPR, the standby RP loads the Cisco IOS software on bootup and initializes itself in standby mode. In the event of a fatal error on the active RP, the system switches to the standby RP, which reinitializes itself as the active RP. In this event, the entire system is rebooted, so the switchover with RPR is slower than with other High Availability switchover features such as Nonstop Forwarding/Stateful Switchover (NSF/SSO).

On the Cisco ASR 1000 Series Router, RPR can also be used to enable a second IOS process on a single RP for a Cisco ASR 1002 or 1004 Router. See the "Second IOS Process on a Cisco ASR 1002 or 1004 Router" section on page 6-5 for additional information on the second IOS process.

For the Cisco ASR 1000 Series Routers, RPR introduces the following functionality:

- Startup configuration synchronization between the active and standby RP or IOS process. It is important to note, however, that changes in the running configuration are not synchronized using RPR.
- Warm Reload—The Warm Reload feature allows users to reload their routers without reading images from storage; that is, the router reboots by restoring the read-write data from a previously saved copy in the RAM and by starting execution without either copying the software from flash to RAM or self-decompression of the image.

It is important to note that in most cases, Stateful Switchover (SSO) requires less downtime for switchover and upgrades than RPR. RPR should only be used when there is a compelling reason to not use SSO.

It is important to note RPR is supported on the Cisco ASR 1000 Series Routers while RPR+ is not.

Stateful Switchover

The Stateful Switchover (SSO) feature takes advantage of processor redundancy by establishing one of the processors as the active processor while the other RP is designated as the standby processor, and then synchronizing critical state information between them. Following an initial synchronization between the two processors, SSO dynamically maintains RP state information between the dual processors.

Stateful Switchover is particularly useful in conjunction with Nonstop Forwarding. SSO allows the dual processors to maintain state at all times, and Nonstop Forwarding lets a switchover happen seamlessly when a switchover occurs.

On the Cisco ASR 1000 Series Router, SSO can also be used to enable a second IOS process on a single RP for a Cisco ASR 1002 or 1004 Router. See the "Second IOS Process on a Cisco ASR 1002 or 1004 Router" section on page 6-5 for additional information on the second IOS process.

It is important to note that in most cases, SSO requires less downtime for switchover and upgrades than RPR. RPR should only be used when there is a compelling reason to not use SSO.

For additional information on NSF/SSO, see the *Cisco Nonstop Forwarding* document.

SSO-Aware Protocol and Applications

SSO-supported line protocols and applications must be SSO-aware. A feature or protocol is SSO-aware if it maintains, either partially or completely, undisturbed operation through an RP switchover. State information for SSO-aware protocols and applications is synchronized from active to standby to achieve stateful switchover for those protocols and applications.

The dynamically created state of SSO-unaware protocols and applications is lost on switchover and must be reinitialized and restarted on switchover.

To see which protocols are SSO-aware on your router, use the following commands **show redundancy client** or **show redundancy history**.

IPsec Failover

IPSec failover is a feature that increases the total uptime (or availability) of a customer's IPSec network. Traditionally, this is accomplished by employing a redundant (standby) router in addition to the original (active) router. If the active router becomes unavailable for any reason, the standby router takes over the processing of IKE and IPSec. IPSec failover falls into two categories: stateless failover and stateful failover.

The IPsec on the Cisco ASR 1000 Series Router supports only stateless failover. Stateless failover uses protocols such as the Hot Standby Router Protocol (HSRP) to provide primary to secondary cutover and also allows the active and standby VPN gateways to share a common virtual IP address.

Bidirectional Forwarding Detection

Bidirectional Forwarding Detection (BFD) is a detection protocol designed to provide fast forwarding path failure detection times for all media types, encapsulations, topologies, and routing protocols. In addition to fast forwarding path failure detection, BFD provides a consistent failure detection method for network administrators. Because the network administrator can use BFD to detect forwarding path failures at a uniform rate rather than the variable rates for different routing protocol hello mechanisms, network profiling and planning is easier, and reconvergence time is consistent and predictable.

On the Cisco ASR 1000 Series Routers, BFD for IPv4 Static Routes and BFD for BGP are supported.

For more information on BFD, see the *Bidirectional Forwarding Detection* document.

Additional References

Related Documents

Related Topic	Document Title
Bidirectional Forwarding Detection	*IP Routing BFD Configuration Guide, Cisco IOS XE Release 3S*
High Availability Configurations	*High Availability Configuration Guide, Cisco IOS XE Release 3S*
Software Upgrade Process Configurations	*Software Upgrade Process Configuration Guide*
Cisco IOS commands	*Cisco IOS Master Commands List, All Releases*

Standards

Standard	Title
None	—

MIBs

MIB	MIBs Link
None	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at this URL: http://www.cisco.com/go/mibs

RFCs

RFC	Title
None	—

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for High Availability Overview

Table 6-4 lists the features in this module and provides links to specific configuration information.

Use Cisco Feature Navigator to find information about platform support and software image support. Cisco Feature Navigator enables you to determine which software images support a specific software release, feature set, or platform. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Note Table 6-4 lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Table 6-4 Feature Information for High Availability Overview

Feature Name	Releases	Feature Information
High Availability Overview	Cisco IOS XE 2.1S	In Cisco IOS XE Release 2.1S, this feature was introduced on the Cisco ASR 1000 Series Router.

Broadband Scalability and Performance

The infrastructure of a service provider must be capable of supporting the services that an enterprise customer or Internet service provider (ISP) wants to offer its subscribers. The service provider must also be able to scale up to an expanding subscriber base. You can configure the Cisco ASR1000 Series Routers for high broadband scalability.

Finding Feature Information in This Module

Your software release might not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the "Feature Information for Broadband Scalability and Performance" section on page 7-11.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Contents

This guide provides information about the following topics:

- PPP Sessions and L2TP Tunnel Scaling, page 7-1
- Configuring the Cisco ASR 1000 Series Router for High Scalability, page 7-3
- Using the cisco-avpair="lcp:interface-config" RADIUS Attribute, page 7-7

PPP Sessions and L2TP Tunnel Scaling

The ASR 1000 Series Routers are deployed in a variety of broadband deployment models for terminating Point-to-Point Protocol (PPP) Sessions and initiating or terminating Layer 2 Tunneling Protocol (L2TP) tunnels. The maximum number of PPP sessions and L2TP tunnels is dependent on the hardware combination. Table 7-1 lists the hardware combinations and the maximum number of PPP sessions and L2TP tunnels that are supported in Cisco IOS XE Release 3.3. For information about the restrictions for PPP sessions and L2TP tunnel scaling, see the "Restrictions for PPP Sessions and L2TP Tunnel Scaling" section on page 7-2.

Table 7-1 Maximum Number of PPP Sessions and L2TP Tunnels Supported on ASR 1000 Hardware

Chassis	RP	ESP	PPP Session	L2TP Tunnel
1001	Integrated	ESP-2.5G or ESP-5G	8000	4000
1001-X	Integrated	2.5G, 5G, 10G, or 20G	8000	4000
1002	Integrated RP1	ESP-5G	12000	6000
1002-X	Integrated	5G, 10G, 20G, or 36G	29000	16000
1004, 1006	RP1	ESP-10G or ESP-20G	24000	12000
1004, 1006	RP2	ESP-20G	32000	16000
1004, 1006, 1013	RP2	ESP-40G	32000[1], 48000, or 64000	16000
1006	RP2	ESP-100G	58000	16000
1013	RP2	ESP-200G	58000	16000

1. 32000 session scaling with RP2 and ESP-40G is fully supported without any additional restrictions. For restrictions on 48000 and 64000 see the "Restrictions for PPP Sessions and L2TP Tunnel Scaling" section on page 7-2.

Table 7-2 lists the virtual circuit limits for ASR 1000 ATM SPAs, effective from Cisco IOS XE Release 3.5.

Table 7-2 ASR 1000 ATM Virtual Circuit (VC) Limits

ASR1000-SIP10		ASR1000-SIP40	
SPA/SIP	**Number of VCs**	**SPA/SIP**	**Number of VCs**
SPA-1XOC12-ATM-V2	4000 VCs per Port, 4000 VCs per SPA	SPA-1XOC12-ATM-V2	4000 VCs per Port, 4000 VCs per SPA
SPA-1XOC3-ATM-V2	4000 VCs per Port, 4000 VCs per SPA	SPA-1XOC3-ATM-V2	4000 VCs per Port, 4000 VCs per SPA
SPA-3XOC3-ATM-V2	3000 VCs per Port, 9000 VCs per SPA	SPA-3XOC3-ATM-V2	4000 VCs per Port, 12000 VCs per SPA
ASR1000-SIP10	24000 VCs per SIP	ASR1000-SIP40	32000 VCs per SIP

Restrictions for PPP Sessions and L2TP Tunnel Scaling

This section lists the restrictions for the PPP sessions and L2TP tunnel scalability.

- To achieve maximum scaling for the 1001, 1002, and 1004 chassis, we recommend software redundancy be disabled for broadband applications, and only hardware redundancy be configured.
- Restrictions on 48000 session scaling with RP2 and ESP-40G include:
  - Intelligent Services Gateway (ISG) Services are not supported

  - Point-to-Point Protocol over ATM (PPPoA) and Point-to-Point Protocol over Ethernet (PPPoEoA) sessions are not supported
  - RP2 requires 16 GB DRAM to support 48000 sessions

Note If any of the listed restrictions is not met, the router scales to a maximum of 32000 sessions only.

- Restrictions on the 64000 session scaling with RP2 and ESP-40G:
  - ISG services are not supported
  - PPPoA and PPPoEoA sessions are not supported
  - Per-session QoS with queuing actions (for example, shaping) is not supported
  - RP2 requires 16 GB DRAM to support 64000 sessions

Note If any of the listed restrictions is not met, the router scales to a maximum of 32000 sessions or 48000 sessions only.

- Restrictions on 64000 L2TP tunnel scaling with RP2 and ESP-40G:
  - ISG services are not supported
  - Per-session QoS is not supported
  - RP2 requires 16 GB DRAM
  - High Availability (SSO) is not supported

Note If any of the listed restrictions is not met, the router scales to a maximum of 16000 L2TP tunnels.

- The RP2 and ESP10 hardware combination is not supported for broadband.
- RP1 with 2GB of DRAM is not recommended for broadband deployment.

Configuring the Cisco ASR 1000 Series Router for High Scalability

The Cisco ASR 1000 Series Routers provide powerful performance and scalability for embedded services.

To achieve maximum scaling on the 1001, 1002, and 1004 chassis, IOS software redundancy must be disabled. For more information about restrictions and limitations, see the "Restrictions for PPP Sessions and L2TP Tunnel Scaling" section on page 7-2.

To ensure high scalability on the Cisco ASR 1000 Series Aggregation Services Router, perform the following configuration tasks:

- Configuring Call Admission Control, page 7-5
- Control Plane Policing, page 7-5

- VPDN Group Session Limiting, page 7-6
- PPPoE Session Limiting, page 7-6

- Monitoring PPP Sessions Using the SNMP Management Tools, page 7-6
- Configuring the Access Interface Input and Output Hold Queue, page 7-6
- Configuring the keepalive Command, page 7-6
- Scaling the L2TP Tunnel Configurations, page 7-7

Configuring Call Admission Control

The Call Admission Control (CAC) feature is configured to protect the ASR 1000 processing resources that must be configured. CAC can restrict the media bandwidth dedicated to active calls when CPU utilization exceeds the configured threshold.

This section provides the following examples for configuring CAC:

Example: Configuring a PPPoE Session

```
router(config)# call admission new-model
router(config)# call admission limit 1000
router(config)# call admission cpu-limit 80
router(config)# call admission pppoe 10 1
```

Example: Configuring a PPPoA Session

```
router(config)# call admission new-model
router(config)# call admission limit 1000
router(config)# call admission cpu-limit 80
router(config)# call admission pppoa 10 1
```

Example: Configuring a VPDN Session

```
router(config)# call admission new-model
router(config)# call admission limit 1000
router(config)# call admission cpu-limit 80
router(config)# call admission vpdn 10 1
```

Control Plane Policing

The Control Plane Policing feature allows you to configure a QoS filter that manages the traffic flow of control plane packets to protect the control plane of Cisco IOS XE routers and switches against reconnaissance and denial-of-service (DoS) attacks. The control plane thus helps maintain packet forwarding and protocol states despite an attack or heavy traffic load on the router or switch.

For examples about configuring the Control Plane Policing feature, see the “Control Plane Policing” section in the *Quality of Service Solutions Configuration Guide* located at:

http://www.cisco.com/en/US/docs/ios-xml/ios/qos_plcshp/configuration/xe-3s/qos-plcshp-ctrl-pln-plc.html

VPDN Group Session Limiting

Using the Virtual Private Dialup Network (VPDN) Group Session Limiting feature, you can limit the number of VPDN sessions allowed per VPDN group. VPDN session limits can increase performance and reduce latency for routers that are otherwise forced to operate at high capacity.

For more information about VPDN Group Session Limiting, see the feature documentation at:

http://www.cisco.com/en/US/docs/ios-xml/ios/vpdn/configuration/xe-3s/vpd-tunnel-mgmt.html

PPPoE Session Limiting

The PPPoE Session Limit Support feature prevents the router from using too much memory for virtual access by limiting the number of PPPoE sessions that can be created on a router or on all Ethernet interfaces and subinterfaces as well as ATM interfaces and subinterfaces.

For more information about PPPoE session limiting, see the feature documentation at:

http://www.cisco.com/en/US/docs/ios-xml/ios/bbdsl/configuration/xe-3s/bba-limit-legcfg-xe.html

Monitoring PPP Sessions Using the SNMP Management Tools

To prevent the virtual access subinterfaces from being registered with the Simple Network Management Protocol (SNMP) functionality of the router and using up the memory, do not use the SNMP management tools of the router to monitor PPP sessions. Use SNMP views to isolate the bulk queries and accidental requests.

Use the **no virtual-template snmp** command to disable the SNMP management tools:

```
Router(config)# no virtual-template snmp
```

Configuring the Access Interface Input and Output Hold Queue

The default value of Gigabit Ethernet and 10 Gigabit Ethernet interfaces is 375 packets for the input and output hold queues. If the interfaces are required to handle a high rate of control packets, such as LCP, IPCP, PPP, L2TP, and DHCP, the default value may not be sufficient. To ensure high scalability, set the access interface input and output hold queue to 4096:

```
Router(config)# interface gig1/0/0
Router(config-if)# hold-queue 4096 in
```

Configuring the keepalive Command

For PPP sessions, the **keepalive** command sets the keepalive timer for a specific interface. To ensure proper scaling and to minimize CPU utilization, set the timer for 60 seconds or longer. The default value is 10 seconds:

```
interface Virtual-Template1
 ip unnumbered Loopback1
 keepalive 60
 no peer default ip address
 ppp authentication pap
```

Note For IP sessions, the keepalives are not enabled by default. Enabling keepalives for IP sessions provides the same capability as PPP keepalives except that ICMP or ARP is used to test the presence of subscribers. For more information about Using ARP for Keepalive Messages and Using ICMP for Keepalive Messages, see the feature documentation at: http://www.cisco.com/en/US/docs/ios-xml/ios/isg/configuration/xe-3s/Configuring_ISG_Policies_for_Session_Maintenance.html

Scaling the L2TP Tunnel Configurations

To prevent head-of-the-line blocking of the IP input process and save system resources, configure the **vpdn ip udp ignore checksum** command:

```
Router(config)# vpdn ip udp ignore checksum
```

When you configure this command, the router directly queues the L2TP Hello packets and Hello acknowledgements to the L2TP control process. We recommend that you configure this command in all the scaled LAC and LNS L2TP tunnel configurations.

If you do not configure the **vpdn ip udp ignore checksum** command, the L2TP software sends the packets to UDP to validate the checksum. When too many packets are queued to the IP input process, the router starts Selective Packet Discard (SPD) mechanism that causes IP packets to be dropped.

Note Head-of-the-line blocking of the IP input process might occur in other nonL2TP configurations. A flush occurring on an input interface indicates that the SPD mechanism is discarding packets.

Using the cisco-avpair="lcp:interface-config" RADIUS Attribute

When you use the **lcp:interface-config** RADIUS attribute to reconfigure the virtual access subscriber interface, call setup rate could be reduced on the Cisco ASR 1000 Series Aggregation Services Routers because the **lcp:interface-config** command syntax includes an IOS interface configuration command. This command is any valid IOS command that can be applied to an interface. When the **lcp:interface-config** attribute is downloaded from the RADIUS server to the Cisco ASR 1000 Series Aggregation Services Routers, the command parser is activated to configure the interface according to AV-pair, determining if the option is valid and then applying the configuration to the virtual access interface (VAI).

The subscriber session scaling on the Cisco ASR 1000 Series Aggregation Services Routers is not impacted by using the lcp:interface-config RADIUS attribute any more than if the equivalent IOS interface command was applied directly onto the virtual-template configuration and was cloned onto the VAI using that method. Using either the lcp:interface-config RADIUS attribute or the virtual-template to apply configuration onto the VAI it is the type of configuration being applied which may in a few cases affect the maximum subscriber session scale of the Cisco ASR 1000 Series Aggregation Services Routers.

Enhancing the Scalability of Per-User Configurations

To enhance scalability of per-user configurations without changing the router configuration, use the ip:vrf-id and ip:ip-unnumbered RADIUS attributes. These per-user vendor-specific attributes (VSAs) are used to map sessions to VRFs and IP unnumbered interfaces. The VSAs are applied to virtual access subinterfaces and are processed during PPP authorization.

The ip:vrf-id attribute is used to map sessions to VRFs. Any profile that uses the ip:vrf-id VSA must also use the ip:ip-unnumbered VSA to install IP configurations on the VAI that is to be created. The PPP that is used on a VAI to be created requires the ip:ip-unnumbered VSA. An Internet Protocol Control Protocol (IPCP) session is not established if IP is not configured on the interface. You must configure either the ip address command or the ip unnumbered command on the interface so that these configurations are present on the VAI that is to be created. However, specifying the ip address and ip unnumbered commands on a virtual template interface is not required because pre-existing IP configurations, if any, are removed when the ip:ip-vrf VSA is installed on the VAI. Therefore, any profile that uses the ip:vrf-id VSA must also use the ip:ip-unnumbered VSA to install IP configurations on the VAI that is to be created.

These per-user VSAs can be applied to VAIs. Therefore, the per-user authorization process does not require the creation of full VAIs, which improves scalability.

Setting the VRF and IP Unnumbered Interface Configurations in User Profiles

Although the Cisco ASR 1000 Series Aggregation Services Router continues to support the lcp:interface-config VSA, the ip:vrf-id and ip:ip-unnumbered VSAs provide another way to set the VRF and IP unnumbered interface configurations in user profiles. The ip:vrf-id and ip:ip-unnumbered VSAs have the following syntax:

```
Cisco:Cisco-AVpair = "ip:vrf-id=vrf-name"
Cisco:Cisco-AVpair = "ip:ip-unnumbered=interface-name"
```

You should specify only one ip:vrf-id and one ip:ip-unnumbered value in a user profile. However, if the profile configuration includes multiple values, the Cisco ASR 1000 Series Aggregation Services Router applies the value of the last VSA received, and creates a virtual access subinterface. If the profile includes the lcp:interface-config VSA, the router always applies the value of the lcp:interface-config VSA.

Setting the VRF and IP Unnumbered Interface Configurations in Virtual Interface Templates

You can specify one VSA value in a user profile on RADIUS and another value locally in the virtual template interface. The Cisco ASR 1000 Series Aggregation Services Router clones the template and then applies the values configured in the profiles it receives from RADIUS, resulting in the removal of any IP configurations when the router applies the profile values.

Redefining User Profiles to Use the ip:vrf-id and ip:ip-unnumbered VSAs

The requirement of a full virtual access interface when using the **lcp:interface-config** VSA in user profiles can result in scalability issues, such as increased memory consumption. This situation is especially true when the Cisco ASR 1000 Series Aggregation Services Router attempts to apply a large number of per-user profiles that include the **lcp:interface-config** VSA. Therefore, when updating your user profiles, we recommend that you redefine the **lcp:interface-config** VSA to the scalable ip:vrf-id and ip:ip-unnumbered VSAs.

The following example shows how to redefine a VRF named *newyork* using the ip:vrf-id VSA:

```
Change:
Cisco:Cisco-Avpair = "lcp:interface-config=ip vrf forwarding newyork"

To:
Cisco:Cisco-Avpair = "ip:vrf-id=newyork"
```

The following example shows how to redefine the Loopback 0 interface using the ip:ip-unnumbered VSA.

```
Change:
Cisco:Cisco-Avpair = "lcp:interface-config=ip unnumbered Loopback 0"

To:
Cisco:Cisco-Avpair = "ip:ip-unnumbered=Loopback 0"
```

Additional References

Related Documents

Related Topic	Document Title
Control Plane Policing	*Quality of Service Solutions Configuration Guide*
VPDN Group Session Limiting	*VPDN Configuration Guide, Cisco IOS XE Release 3S*
PPPoE session limiting	*Configuring PPP over Ethernet Session Limit Support Feature Guide*
Using ARP for Keepalive Messages and Using ICMP for Keepalive Messages	*Intelligent Services Gateway Configuration Guide Cisco IOS XE Release 3S*
Cisco IOS commands	*Cisco IOS Master Commands List, All Releases*

Standards

Standard	Title
None	—

MIBs

MIB	MIBs Link
None	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at this URL: http://www.cisco.com/go/mibs

RFCs

RFC	Title
None	—

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for Broadband Scalability and Performance

Table 7-3 lists the features in this module and provides links to specific configuration information.

Use Cisco Feature Navigator to find information about platform support and software image support. Cisco Feature Navigator enables you to determine which software images support a specific software release, feature set, or platform. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Note Table 7-3 lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Table 7-3 Feature Information for Broadband Scalability and Performance

Feature Name	Releases	Feature Information
High Availability Overview	Cisco IOS XE 2.1S	In Cisco IOS XE Release 2.1S, this feature was introduced on the Cisco ASR 1000 Series Router.
Walk-by User Support for PWLAN in ISG	Cisco IOS XE 3.7S	In Cisco IOS XE Release 3.7S, this feature was introduced on the Cisco ASR 1000 Series Router.

Multilink PPP Support for the Cisco ASR 1000 Series Routers

First Published: October 2012
Last Updated: May 25, 2015

Multilink Point-to-Point Protocol (MLP) provides support to aggregate the bandwidth of low-speed WAN and broadband links into a single entity, referred to as a bundle interface. A bundle interface is a logical entity that provides a single point in which other features (for example, Quality of Service [QoS]) can be attached. MLP provides incremental bandwidth on demand, by adding additional links to the bundle, as needed. MLP also enables interleaving of latency-sensitive priority traffic with fragmented nonpriority traffic using link fragmentation and interleaving (LFI).

Member links that are a part of an MLP bundle can be bundled across ports on:

- The same shared port adapter (SPA)
- Different SPAs on the same SPA interface processor (SIP)
- Different SPAs on different SIPs

The Cisco IOS XE software supports MLP links for serial (T1, E1, NxDS0) and broadband topologies such as Multilink PPP over ATM (MLPoA), Multilink PPP over Ethernet (MLPoE), Multilink PPP over Ethernet over ATM (MLPoEoA), and Multilink PPP over LNS (MLPoLNS). Additionally, the Cisco IOS XE software allows the device to operate as an L2TP Access Concentrator (LAC), L2TP Network Server (LNS), or PPP Termination and Aggregation (PTA) device.

This document describes the features, limitations, and scaling of MLP on the Cisco ASR 1000 Series Aggregation Services Routers running the Cisco IOS XE software. For information about the configuration and operation of MLP in the Cisco IOS XE software, see the "Configuring Multilink PPP Connections" chapter in the *Wide-Area Networking Configuration Guide: Multilink PPP, Cisco IOS XE Release 3S (Cisco ASR 1000)*.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest information about features and caveats, see the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the ?$paranum>Feature Information for Multilink PPP Support for Cisco ASR 1000 Series Routers? section on page 8-18.

Use the Cisco Feature Navigator to find information about platform support and Cisco IOS and Cisco IOS XE operating system software image support. To access the Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Not all commands may be available in your Cisco IOS software release. For release information about a specific command, see the corresponding command reference documentation.

Contents

- Cisco IOS XE Scaling Limits for MLP Bundles, page 8-3
- Restrictions for MLP over Serial Interfaces, page 8-5
- Restrictions for MLP over Ethernet at PTA and LAC, page 8-5
- Restrictions for MLP over ATM at PTA and LAC, page 8-7
- Restrictions for MLP at LAC, page 8-7
- Restrictions for MLP over LNS, page 8-7
- Information About Multilink PPP Support for Cisco ASR 1000 Series Routers, page 8-8
- , page 8-9
- Quality of Service, page 8-9
- Bandwidth, page 8-12
- MTU, page 8-12
- Downstream LFI, page 8-13
- IP Type of Service Reflect, page 8-15
- IP Tunnel Marking, page 8-15
- Unsupported Features, page 8-16
- Additional References, page 8-16
- Feature Information for Multilink PPP Support for Cisco ASR 1000 Series Routers, page 8-18

Cisco IOS XE Scaling Limits for MLP Bundles

This section lists the scaling limits for MLP bundles in different releases of Cisco IOS XE, in which scaling limits were either introduced or enhanced.

Release 2.2.(0)S

In Cisco IOS XE Release 2.2.(O)S, the MLP feature was introduced on the Cisco ASR 1000 Series Aggregation Services Routers. MLPoSerial was the first supported transport. In this release, MLP bundles can consist of up to 10 serial links. The bandwidth of each link interface does not have to be the same as the other links in the bundle. The Cisco ASR 1000 Series Aggregation Services Routers support links of types T1, E1, and NxDS0. MLP LFI is fully supported with MLPoSerial in this release.

Release 3.4.(0)S

In Cisco IOS XE Release 3.4.(O)S, the MLP feature was enhanced to enable the Cisco ASR 1000 Series Aggregation Services Routers to act as LAC, LNS, or PTA devices. Support for tunneling bundles between the LAC device and the LNS device was added. In this release, transport between the LAC device and the LNS device is Layer 2 Tunnel Protocol (L2TP). The L2TP tunnels can operate on either 1-Gbps or 10-Gbps interfaces. When ASR 1000 Series Aggregation Services Router acts as an LNS device, it terminates the MLP bundles coming through the L2TP tunnel from the LAC. In this release, support was added for MLP upstream fragment reassembly, but not for MLP downstream fragmentation.

Release 3.7.1S

In Cisco IOS XE Release 3.7.1S, the existing support for the MLP feature in a broadband topology was enhanced. The scaling limits were increased for the Ethernet transports, and downstream fragmentation support was added for the broadband topologies.

In this release, when a Cisco ASR 1000 Series Aggregation Services Router acts as an LNS device, it terminates the MLP bundles coming through the L2TP tunnel from the LAC. The scaling targets mentioned for MLP over broadband are based on RP2/ESP40 and 2RU-VE hardware configurations. The scaling capabilities are less for RP1 and ESP5, ESP10, or ESP20.

The implementation of MLP on a Cisco ASR 1000 Series Aggregation Services Router does not support all the Cisco IOS XE interoperability features.

Release 3.12.(0)S

In Cisco IOS XE Release 3.12.(O)S, the multi-member-link MLPoA or MLPoEoA, including Downstream , is introduced. The scaling limits are increased for the member links in MLPoA or MLPoEoA scenarios.

Table 8-1 shows the maximum scale numbers for various MLP functionalities on the Cisco ASR 1000 Series Aggregation Services Routers.

Table 8-1 MLP Features and Maximum Scale Numbers

Transport	Maximum Number of Members per Bundle	Maximum Number of Bundles per System	Maximum Number of Member Links per System	Downstream LFI	Upstream Fragment Reassembly	Cisco IOS XE Release
MLPoSerial	10	1232	1232	Yes	Yes	2.2.0S
MLPoA AAL5MUX	1	1000	1000	No	Yes	3.4.0S
MLPoA AAL5MUX	8	4000	4000	Yes	Yes	3.12.0S
MLPoA AAL5SNAP	1	1000	1000	No	Yes	3.4.0S
MLPoA AAL5SNAP	8	4000	4000	Yes	Yes	3.12.0S
MLPoE	1	4000	4000	No	Yes	3.4.0S
MLPoE	8	4000	4000	Yes	Yes	3.7.1S
MLPoEoA AAL5SNAP	1	1000	1000	No	Yes	3.4.0S
MLPoEoA AAL5SNAP	8	4000	4000	Yes	Yes	3.12.0S
MLPoEoQinQ	1	4000	4000	No	Yes	3.4.0S
MLPoEoQinQ	8	4000	4000	Yes	Yes	3.7.1S
MLPoEoVLAN	1	4000	4000	No	Yes	3.4.0S
MLPoEoVLAN	8	4000	4000	Yes	Yes	3.7.1S
MLPoLNS	1	4000	4000	No	Yes	3.4.0S
MLPoLNS	8	4000	4000	Yes	Yes	3.7.1S

Restrictions for MLP over Serial Interfaces

The following restrictions apply to MLP over Serial Interfaces:

- The MLP over Serial Interfaces feature supports a maximum of ten member links per bundle. The member links can be any combination of T1/E1 or fractional T1s/E1s (for example, NxDS0). Member-link interface speed above T1/E1 is not supported in the MLP over Serial Interfaces feature. For better MLP performance, all the member links in a bundle must be of the same bandwidth.
- Member links in a bundle cannot be of different encapsulation types.
- You cannot manually configure the bandwidth of an MLP bundle by using the **bandwidth** command on the multilink interface. The bandwidth of an MLP bundle is managed based on the aggregate bandwidth of all the active member links on the bundle. As the links are dynamically added or removed from an MLP bundle, the bandwidth is updated to reflect the aggregate of the active links. The bandwidth can be rate limited by applying an hierarchical QoS policy on the multilink interface and applying a shaper to the parent class-default class.
- MLP over Frame Relay is not supported; only MLP over Serial PPP link is supported. Customers who require multilink support in a frame relay environment can use the Multilink Frame Relay (MLFR-FRF.16) feature.
- The legacy IOS compression feature **compress** [**mppc** | **stac** | **predictor**] is not supported.
- LFI is supported on MLP bundles with any number of links in the bundle. However, when using a bundle with more than one member link, the order of the priority packets (PPP encapsulated) is not guaranteed. Priority-packet distribution is handled in a manner similar to IP per-packet load sharing. MLP guarantees nonpriority packet ordering that manages reordering at the peer device, based on the MLP packet sequence number.
- Order issues with the LFI multiple-member link in case of priority traffic can be addressed in some platforms using Multiclass Multilink Protocol (MCMP-RFC 2686), which is an extension of the MLP. The Cisco ASR 1000 Series Aggregation Services Routers do not support MCMP.
- Only the MLP long-sequence number format is supported for the packet header format option.

Restrictions for MLP over Ethernet at PTA and LAC

The following restrictions apply to MLP over Ethernet at PTA and LAC:

- MLPoE using EtherChannel is not supported.
- For MLP virtual access bundles, the default Layer 3 (that is IP and IPv6) maximum transmission unit (MTU) value is 1500. For more information about MTU, see MTU, page 8-12.
- For MLPoE PTA variations (MLPoE, MLPoVLAN, and MLPoQinQ), the default bandwidth of the member-link session is 1 Gbps instead of the data rate communicated by the DSLAM to the PTA router. If a bandwidth statement is added to the virtual template, the bandwidth is applied to the bundle instead of the member link. This is not the desired behavior. (To define the data rate of an MLPoE PTA-type bundle, apply a QoS policy on the bundle session that includes a parent shaper on the class-default class with an explicit data rate defined. Do not use the **shape percent** command in this parent shaper because the **shape percent** command uses the default data rate of 1 Gbps as the base rate for percent calculation. However, the percent-based rates can be defined in the child (nested) policy, if an hierarchical policy is being defined.

- If the DSLAM between the CPE and PTA communicates the link rate through the PPPoE dsl-sync-rate tags (Actual Data-Rate Downstream [0x82/130d] tag), the PTA device passes this data to the RADIUS server, but the Cisco ASR 1000 Series Aggregation Services Routers do not act upon it. The data rate of the session remains as described in the previous list item.

Restrictions for MLP over ATM at PTA and LAC

The following restrictions apply to MLP over ATM at PTA and LAC:

- ATM Autosense is supported to allow the dynamic selection of MLPoA or MLPoEoA.
- For ATM, the link-level bandwidth is a part of the ATM Permanent Virtual Circuits (PVC) configuration based on the unspecified bit rate (UBR) or variable bit rate (VBR) configurations. The bundle bandwidth is the aggregate of the member-link session bandwidth.

Note The MLP over Ethernet over ATM at PTA and LAC has the same restrictions as the MLP over ATM at PTA and LAC.

Restrictions for MLP at LAC

In case of MLP over LNS (Ethernet) LAC switching, the MLP member-link session and the packet payload is transparent at the LAC device because it does not terminate the MLP session or the bundle interface. Hence, the LAC device does not bind the number of member-link sessions associated with a bundle. Similarly, the LFI functionality is transparent at the LAC device because the traffic is switched or passed through traffic.

Restrictions for MLP over LNS

The following restrictions apply to MLP over LNS:

- MLPoLNS bundles are supported with only Ethernet as the trunk between the LAC and LNS.
- Layer 2 Tunnel Protocol (L2TP) over IPsec is not supported.
- QoS (other than downstream Model-F shaping) on interfaces and tunnels towards the customer premise equipment (CPE) is not supported.
- When the CPE client initiates the PPP LCP connection, the multilink negotiation included as part of the LCP negotiation may fail if the LAC has not yet established connection with the LNS (which is typically the case). The LNS renegotiates the Multilink LCP options with the CPE client when the LAC initiates the connection to the LNS. (To allow this renegotiation of LCP options, the **lcp renegotiation always** command must be configured in the VPDN group at the LNS).
- Although per-packet load balancing is not supported, the configuration is not blocked and the functionality is operational (but not tested). Per-packet load balancing cannot be used with MLPoLNS because MLPoLNS requires a single-path per-destination IP address.
- Unlike the MLP over Serial mode or the MLP PTA mode, packets may traverse several network hops between the CPE and LNS devices in an MLPoLNS network. As a result of this multihop topology, even on a single-link bundle, MLP encapsulated packets may arrive at the receiver in an out-of-order state. Hence, the MLPoLNS receiver operates in a loose, lost-fragment detection mode. In this mode, if an MLP fragment is lost, the received MLP waits for a short time to receive the lost fragment. In addition, the MLP receiver limits the amount of out-of-order MLP data received before the fragment is declared lost. In Cisco IOS XE software, the default timeout value is 1 second. This may create problems in an environment with high packet loss and scaled MLP configurations because it requires the receiver to potentially buffer large amounts of data for each MLP bundle. Since the buffer space that is available is a finite resource, worst-case depletion of buffers can bleed

over and begin affecting packet buffering on other MLP bundles. (The MLP lost-fragment timeout can be configured on the multilink virtual template interface using the **ppp timeout multilink lost-fragment** (*seconds*) (*milliseconds*) configuration command).

By default, in MLPoLNS, the Cisco IOS XE software informs the MLP that packets may arrive out of order. This works well for upstream traffic, but does not address the order issue at the peer CPE device. The peer CPE device should also be configured to allow for receipt of out-of-order packets. In Cisco devices, this can be managed by configuring the **ppp link reorders** command at the bundle interface.

- When the Cisco ASR 1000 Series Aggregation Services Routers function as both a PTA device and an LNS device simultaneously, locally terminated member links (PTA) and member links that are forwarded from the LAC are not supported within the same bundle.

Restrictions for Broadband MLP at PTA and LNS

The following restrictions apply to all variations of broadband MLP at PTA and LNS modes:

- When defining an MLP bundle with multiple member-link sessions, we recommend that all the member-link sessions utilize the same physical interface or subinterface. If other broadband sessions are sharing the same interface, ensure that all the member-link sessions utilize the same physical interface or subinterface.
- The following issues might occur because of splitting links across separate physical interfaces or subinterfaces:
  - MLP is a sequenced protocol and all the packets and fragments must be reordered and reassembled at the receiver, based on the MLP sequence number before the receiver forwards them. In such a scenario, packets traversing separate physical interfaces may cause additional packet latency disparity between links due to transmission delays and other issues associated with using multiple physical paths. The reordering and reassembly processing may require additional MLP buffering at the receiver.
  - MLP on the Cisco ASR 1000 Series Aggregation Services Routers performs congestion management of the MLP bundle based on the congestion state of the member-link sessions that make up the bundle. If member-links are distributed across multiple interfaces and sufficient congestion is detected in one or more member links, the bundle may be back pressured due to the congestion even if all the links in the bundle are not congested. By keeping all the links on the same physical interface or subinterface, the chance of back pressure due to one link being congested is reduced.

Information About Multilink PPP Support for Cisco ASR 1000 Series Routers

The Multilink PPP feature provides the load-balancing functionality over multiple WAN links, while providing multivendor interoperability, packet fragmentation, proper sequencing, and load calculation for both inbound and outbound traffic. Cisco implementation of MLP supports the fragmentation and packet-sequencing specifications described in RFC 1990.

- Quality of Service, page 8-9
- Bandwidth, page 8-12
- MTU, page 8-12

- Downstream LFI, page 8-13
- IP Type of Service Reflect, page 8-15
- IP Tunnel Marking, page 8-15

Some Cisco IOS platforms use the **interface multilink** command for both MLP over Serial and MLP over ATM (MLPoA) to configure multilink bundle interfaces. On the Cisco ASR 1000 Series Aggregation Services Routers, multilink bundle interfaces are configured using the **interface multilink** command for MLP over Serial and the **interface Virtual-Template** command for MLPoA.

On the Cisco ASR 1000 Series Aggregation Services Routers, all broadband MLP configurations use the **interface Virtual-Template** command to define the multilink bundle configuration. A virtual access interface is created dynamically from the virtual template when the session is negotiated with the peer device.

Quality of Service

QoS refers to the ability of a network to provide improved service to selected network traffic over various underlying technologies, including Frame Relay, ATM, Ethernet and 802.1 networks, SONET, and IP-routed networks. In particular, QoS features provide improved and more predictable network service.

For serial deployments, QoS is applied to an MLP bundle using the **multilink** configuration command. For broadband deployments, QoS is applied to an MLP bundle using the **virtual-template** command. When a router dynamically creates the virtual access interface from the virtual template, the QoS policy is applied to the corresponding bundle.

QoS is characterized by the following features and restrictions:

- To rate limit a broadband MLP bundle session, use a hierarchical QoS (HQoS) policy with a parent shaper in the class-default class.
- The Cisco ASR 1000 Series Aggregation Services Routers support HQoS queuing only in the egress (output) direction, and not in the ingress direction.
- The Cisco IOS XE software supports Model-F QoS with MLP.

Note Model-F QoS on the L2TP tunnel is not supported on the Cisco ASR 1002-X Router and the FP100 line card.

  - In Cisco IOS XE Release 3.7.1S, support was added for Model-F QoS on the L2TP tunnel when the device acts as an LNS. A parent shaper policy can be applied to the physical subinterface that connects the LNS to the LAC device. This enables the shaping of the aggregate traffic going downstream to the LAC device.
  - If a Model-F shaper is attached to the LAC-facing interface after the sessions are established through that interface, the sessions must be bounced to handle the priority traffic appropriately.
- In Cisco IOS XE Release 3.4S, the **shape average** *shape-rate* **account user-defined** <*-63 to 63*> **[atm]** command supports only the broadband MLP interface and not the MLP over Serial interface. In Cisco IOS XE Release 3.6S, the **shape average** *shape-rate* **account user-defined** <*-63 to 63*> **[atm]** command also supports MLP over Serial Interface.
- ATM cell loss priority (CLP) Match (classification) and Set (marking) are not supported with broadband MLP.

- When packets transit the MLP transmit path, they are subject to two separate stages of queuing. The first stage is at the MLP bundle interface, where QoS may be applied, and the second one is at the MLP member-link interface. At the MLP bundle interface, the packets are processed according to the applied QoS policy. Packets classified as priority are given preferential treatment over nonpriority traffic.

  For the priority classification to be honored at the MLP member-link interface, the bundle must have **ppp multilink interleave** enabled. Interleaving allows a packet to be queued to a separate priority queue at the member-link. If interleaving is not enabled on the bundle, the priority packet is placed in the member link session default queue and the knowledge that it is a priority packet will be lost. This is especially important if there are other PPP or MLP sessions sharing the same physical interface or subinterface. Without interleaving, priority traffic on the other sessions are given preferential treatment over the MLP priority packets that were reclassified as nonpriority packets at the MLP member-link queuing stage. For additional information on interleaving, see the Downstream LFI, page 8-13.

Multilink PPP Packet Overhead Accounting for Shaping and Policing

On the Cisco ASR 1000 Series Aggregation Services Routers, Multilink PPP adjusts the packet length presented for shaping and policing to include the additional Layer 2 overhead added by Multilink PPP. For MLP over Serial, overhead accounting includes the MLP and PPP Layer 2 overhead. For Broadband MLPs such as MLPoE, MLPoEoVLAN, MLPoEoQinQ, MLPoEoA, MLPoA, and MLPoLNS, overhead accounting includes the MLP, PPP, Ethernet, ATM, and L2TP (LNS) Layer 2 overhead. If the output interface is ATM, such as the MLPoA or MLPoEoA, the Cisco ASR 1000 Series Aggregation Services Routers also take into account the ATM Cell overhead for the shaper. The ATM Cell overhead is not accounted for policing.

Shaping and policing overhead accounting does not include the additional overheads added by a SPA such as, Ethernet CRC, preamble, IPG, serial interface CRC, start of packet (SOP) delimiter, end of packet (EOP) delimiter, and serial-bit stuffing (the only exception being the ATM Cell overhead for the shaper referred to earlier). The overhead added by a SPA can be included in the shaper using the QoS **shape accounting user-defined** option.

If you do not define a QoS shaper for the multilink bundle interface, a default shaper is applied to the bundle based on the aggregate bandwidth of all the links that make up the multilink bundle. The information contained in this section applies to both the default shaper and a QoS user-defined shaper, which the user may explicitly configure and apply to a multilink bundle.

The priority packets that are interleaved are sent PPP encapsulated and the MLP Layer 2 overhead is not included because MLP encapsulation is not included in these packets. During overhead accounting for link fragmentation, overhead accounting calculations are performed prior to the actual link fragmentation and link selection for Multilink PPP load balancing.

If all the member links in the corresponding multilink bundle use the same fragment size, the number of fragments are calculated and the overhead is adjusted to include the additional per-fragmentation Layer 2 header overhead for the shaper and policer. If one or more links in the bundle use different fragment sizes, the number of fragments cannot be calculated with 100 percent accuracy because link selection for load balancing and fragment size is not known until QoS processing is completed at the bundle level (after shaping and policing). For links with unequal fragment size, a *best effort* attempt is made using the largest link fragment size on the bundle. By using the largest fragment size, MLP avoids undersubscribing the member-link interfaces. If the links become oversubscribed, MLP will backpressure the bundle to avoid sustained oversubscription of the member links.

In Cisco IOS XE Release 3.4S on the Cisco ASR 1000 Series Aggregation Services Routers, support for shaping and policing overhead accounting was added for Broadband Multilink PPP. In addition, support was added for the Shape User-Defined Overhead Accounting feature using the following QoS command:

shape [average | peak] mean-rate [*burst-size*] [*excess-burst-size*] **account {{{qinq | dot1q} {aal5 | aal3} {subscriber-encapsulation}} | {***user-defined offset* **[atm]}}**

This command enables you to include the additional overhead added by a SPA using the **user-defined** option. For example, the Ethernet SPA adds an additional 24 bytes per packet so that a user-defined value of 24 covers Ethernet IPG (12) + Preamble (8) + CRC32 (4). Another interesting scenario is when deploying MLPoLNS in an ATM topology. The physical link between the LNS and the LAC is Ethernet, and the physical link between the LAC and the CPE is ATM. In such a scenario, you can add the **atm** keyword to include the ATM Cell overhead between the LAC and the CPE.

In Cisco IOS XE Release 3.6S, shaping and policing overhead accounting support was added for Serial Multilink PPP and Multilink PPP LFI.

For more information on shaping and policing, see the *IOS XE Ethernet Overhead Accounting documentation* at:
http://www.cisco.com/en/US/docs/ios-xml/ios/qos_plcshp/configuration/xe-3s/qos-plcshp-ether-ohead-actg.html

Downstream Model-F Shaper on LNS

From Cisco IOS XE Release 3.7.1S, Model-F downstream shaping support for MLPoLNS is available to the Cisco ASR 1000 Series Aggregation Services Routers when these routers function as an LNS device.

Note Model-F downstream shaping for MLPoLNS is not supported on the Cisco ASR 1002-X Router and the FP100 line card.

This section provides an example of a Model-F policy with a parent shaper policy attached to a VLAN interface on the LNS device. The VLAN interface is used for the L2TP tunnel between the LAC device and the LNS device. The following configuration example shows an aggregate shaper applied to a VLAN, which shapes all the MLP sessions going downstream to the LAC device:

```
policy-map lns_downstream_shaper_out
    class class-default
        shape average 5000000

interface GigabitEthernet0/1/0.2
    encapsulation dot1Q 2
    ip address 90.0.0.1 255.255.255.0
    service-policy output lns_downstream_shaper_out
```

Note Model-F QoS allows a parent shaper on the class-default class by using a flat policy. No additional QoS functionalities are supported in the Model-F policy.

Bandwidth

The interface-level **bandwidth** command must not be used to define the bandwidth at the bundle level on the virtual template interface or the multilink interface. By default, the bundle bandwidth is the aggregate of the bandwidth of the individual member links that make up the bundle.

For ATM, the link-level bandwidth is part of the ATM Permanent Virtual Circuits (PVC) configuration based on the unspecified bit rate (UBR) or variable bit rate (VBR) configurations. The member-link bandwidth cannot be set for an MLPoE session on a PTA device. To define the bandwidth for an MLPoE-type bundle on a PTA device, a QoS policy must be applied to the bundle interface that shapes the bundle bandwidth at the class-default class with a parent shaper.

In PPPoE and MLPoE broadband networks, the DSL access multiplexer (DSLAM) placed between the customer premises equipment (CPE) and LAC or PTA, inserts a PPPoE vendor tag. This tag includes information such as, media rate, characteristics, and identification pertaining to the circuit or session.

For more information about Ethernet-based networks, see *DSL Forum TR-101 Migration to Ethernet-Based DSL Aggregation April 2006* at:

http://www.broadband-forum.org/technical/download/TR-101.pdf

The PTA passes media-rate information to the RADIUS server for selecting an appropriate QoS policy to the bundle session based on the reported bandwidth. In the context of MLP over LNS, the LAC passes media-rate information to both the RADIUS server and the LNS router. The LNS router uses the media-rate information to define the bandwidth of the corresponding member-link session. If the upstream connection at the LAC is MLPoE, MLPoEoVLAN, or MLPoEoQinQ, the DSLAM may provide the media rate information to the LAC. If the DSLAM does not provide the media rate, the member-link session bandwidth can be configured using the **l2tp tx-speed** *rate* and **l2tp rx-speed** *rate* commands within the **vpdn-group** configuration command or downloaded from the RADIUS server using the l2tp-tx-speed and l2tp-rx-speed attributes.

MTU

For MLP Virtual Access bundles (IP and IPv6), the default Layer 3 MTU value is 1500. When the MLP bundle's member links are Ethernet, as in MLPoE, MLPoEoVLAN, and MLPoEoQinQ, the default MTU value of 1500 may cause an issue when sending IP packets that are close to this size.

For example, when a router sends a 1500-byte IP packet over MLPoE, the actual packet size transmitted is 1528: 14 (Ethernet header) + 8 (PPPoE header) + 6 (MLP header) + 1500 (IP) = 1528. The peer router drops the incoming packet as a **giant** because it exceeds the default expected maximum packet size.

The 1500-byte MTU size does not take into account any PPPoE or MLP header overhead, and hence, causes packets greater than 1493 bytes to be dropped by the peer.

To address this issue, perform one of the following tasks:

- Lower the MTU on the MLP bundle to 1492.
- Increase the MTU on the Ethernet interface to 9216, the maximum MTU size that the Cisco ASR 1000 Series Aggregation Services Routers support.

Note In the Cisco ASR 1000 Series Route Processor 1 (RP1), 2RU, and 2RU-Fixed chassis, the MTU size for the Management Ethernet interface (interface gigabitethernet 0) is limited to 2370 bytes.

Downstream LFI

Although LFI is thought of as a single feature, it is actually two independent features within MLP. MLP link fragmentation allows larger packets to be Layer 2 fragmented by MLP, and the fragments to be distributed across the various member links in the MLP bundle. These fragments are MLP encapsulated and sequenced. These fragments are then collected, reordered, and reassembled at the peer termination point for the MLP bundle interface.

Note For more information about interleaving with QoS, see ?$paranum>Quality of Service? section on page 8-9.

Interleaving enables you to reduce transmission delay on delay-sensitive voice, video, and interactive application data by interleaving it with the MLP fragments. When interleaving is configured, the packets on the bundle interface that QoS classifies as priority packets are interleaved. These priority packets are PPP encapsulated and interleaved with the MLP-encapsulated fragments or packets. When the peer router receives the PPP packets, they can be immediately forwarded, whereas, the received MLP-encapsulated packets have to be reordered and reassembled before being forwarded. While link fragmentation and interleaving can be configured on any multilink bundle, this LFI functionality is beneficial only on bundles of 1 Mbps or less. Packet transmission delays of higher bandwidth bundles are such that QoS prioritization of priority traffic should be sufficient to guarantee preferential treatment of the priority traffic without the need for LFI.

One downside of interleaving is that when there are two or more links in an MLP bundle, the order of the PPP-encapsulated packets cannot be guaranteed. In most applications sending data, such as, voice, video, and Telnet, this is not an issue because the gap between the packets on a given flow is large enough that the packets must not pass each other on the multiple links in the bundle. Since the order cannot be guaranteed for the priority PPP-encapsulated packets that are interleaved, IP Header Compression (IPHC) is skipped on any packet that is classified as priority-interleaved packet. IPHC continues to occur for nonpriority packets that are sent as MLP encapsulated because MLP guarantees reordering before the packets are forwarded to IPHC.

The Multi-Class Multilink Protocol (MCMP) (RFC-2686) addresses the issues related to ordering of priority-interleaved packets. Currently, the MCMP is not supported on the Cisco ASR 1000 Series Aggregation Services Routers.

MLP LFI must be configured on the Cisco ASR 1000 Series Aggregation Services Routers to enable LFI.

In the context of interface multilink or interface virtual template, use any of the following commands to enable link fragmentation:

- **ppp multilink fragment delay** (delay in milliseconds)
- **ppp multilink fragment size** (maximum fragment size, in bytes)
- **ppp multilink interleave**

For MLP using serial links, link fragmentation can also be enabled by configuring the **ppp multilink fragment size** (maximum fragment size, in bytes) command on the member-link serial interface.

If the MLP bundle has only one active member link and interleaving is not enabled, MLP fragmentation is disabled. In addition, all the packets are sent PPP encapsulated instead of MLP encapsulated. When a second link in the bundle becomes active or interleaving is enabled, MLP and fragmentation is enabled.

If the **ppp multilink interleave** command is not configured, only MLP link fragmentation is enabled. To enable interleaving, you must also configure the **ppp multilink interleave** command at the interface multilink level or the interface virtual template level. In addition to configuring interleaving as indicated

here, you must also define a QoS policy with one or more priority classes, and attach the QoS to this interface using the **service-policy output** *policy-map name* command. This command classifies the priority traffic, that is interleaved by the MLP.

See the QoS and LFI configuration examples in the "Configuring Multilink PPP Connections" chapter in the *Wide-Area Networking Configuration Guide: Multilink PPP, Cisco IOS XE Release 3S (Cisco ASR 1000)*.

When configuring MLP fragmentation on the various Cisco platforms, the functionality of MLP fragmentation and interleaving support on the various platforms may differ. This section explains the configuration options and their interpretation in the context of the Cisco ASR 1000 Series Aggregation Services Routers.

Based on the values of the MLP fragmentation configuration commands, the MLP feature calculates two values that are used during MLP fragmentation: link weight and maximum fragment size. These parameters are calculated for each member link in the bundle.

First, a link weight must be determined for each member link. The link weight indicates the number of bytes, and the MLP uses this value to balance the data amongst the links in the bundle. This parameter is especially important when the links in a bundle are of unequal bandwidth. The link weight is based on a combination of the bandwidth of the member link and the PPP multilink fragment delay value. If you do not configure the fragment delay value, a default delay value of 30 milliseconds is used:

Link Weight = (Member Link Interface Bandwidth in bps/8) * Fragment Delay

Caution Configuring the fragment delay to a smaller value results in smaller fragment size because the fragment delay value determines the default fragment size on the member link. This, in turn, implies loss of bandwidth due to the added Layer 2 header overhead. This is important for broadband MLP, which can have Layer 2 headers of 4 to 58 bytes in length.

The default maximum fragment size must be calculated per member link. The default maximum fragment size used will be the lesser value obtained from either of the following calculations:

- Link Weight – Multilink PPP + PPP Header Overhead (8)
- Interface MTU – Multilink PPP Header Overhead (4)

After the default maximum fragment size is calculated, if you have configured the **ppp multilink fragment size** (maximum) command at the multilink, virtual template, or serial interface level, the default maximum fragment size is compared against the configured maximum value and is capped accordingly. If the fragment size is configured at the serial interface level and the multilink interface level, the serial interface configuration takes precedence.

MLP Fragmentation Model

Earlier, some Cisco platforms supported a legacy MLP fragmentation model that was enabled by default if all the following criteria were met:

- Two or more active member links exist in the bundle.
- All the member links have equal bandwidth.
- No other form of multilink fragmentation or interleave commands are configured on the bundle or member-link interface.

In the legacy model, there were many instances when fragmentation was enabled by default without users being aware that it was configured. In addition, packets of moderate length could be fragmented. This did not provide the expected throughput on the bundle due to the added packet Layer 2 overhead introduced by MLP fragmentation.

On the Cisco ASR 1000 Series Aggregation Services Routers, this model of MLP fragmentation was supported until Cisco IOS XE Release 3.7.0. Effective from Cisco IOS XE Release 3.7.1, the Cisco ASR 1000 Series Aggregation Services Routers do not support this mode of MLP fragmentation. Therefore, you must now explicitly configure the multilink fragmentation or interleaving to enable MLP fragmentation.

Effective from Cisco IOS XE Release 3.7.1, the following MLP configuration commands are ignored by the Cisco ASR 1000 Series Aggregation Services Routers:

- **ppp multilink fragment disable**
- **ppp multilink fragment maximum** *maximum number of fragments per packet*

IP Type of Service Reflect

Effective from Cisco IOS XE Release 3.7.(0)S, support for the IP Type of Service (ToS) Reflect feature was added on the VPDN group or VPDN template for the L2TP tunnel when the Cisco ASR 1000 Series Aggregation Services Routers act as LNS devices for broadband MLP sessions. Later, this feature was also added to the following maintenance releases: Cisco IOS XE 3.4.2, 3.5.1, and 3.6.2.

The IP Type of Service (ToS) Reflect feature allows the IP header ToS value from the inner IP header to be reflected in the ToS of the outer L2TP IP header.

Caution To prevent MLP packet reordering and fragment or packet holes, the ToS data should not be used to reclassify and requeue or drop packets at the LAC. Any drops or reordering of MLP packets may cause MLP reordering or reassembly delays and additional packet loss in the receiving CPE device.

The following example shows how to configure IP ToS reflect:

```
vpdn-group vpdn-1
    accept-dialin
        protocol l2tp
        virtual-template 1
    session-limit 100
    terminate-from hostname VPDN-1
    lcp renegotiation always
    no l2tp tunnel authentication
    ip tos reflect
```

IP Tunnel Marking

Effective from Cisco IOS XE Release 3.7.1, support was added for setting the ToS value in the outer L2TP IP header using the QoS set tunnel action or the policer set tunnel action.

The following configuration options of the set actions are supported when applied to the output QoS policy of the multilink virtual template interface. This functionality is not supported in the Model-F QoS policy attached to the member-link parent subinterface.

- **set ip dscp tunnel** *xx*
- **set ip prec tunnel** *xx*

- **set dscp tunnel** *xx*
- **set prec tunnel** *xx*
- **police set-dscp-tunnel-transmit** *xx*
- **police set-prec-tunnel-transmit** *xx*

The following example shows how to set the ToS value using the **police set-prec-tunnel-transmit** option:

```
policy-map ppp
    class class-default
        police cir 4000000 conform-action set-prec-tunnel-transmit 3

Set action example:

policy-map ppp
    class gold
        set ip prec tunnel 1
```

Unsupported Features

The Cisco ASR 1000 Series Aggregation Services Routers do not support the following MLP features:

- In-Service Software Upgrade (ISSU) and Stateful Switchover (SSO) for MLP bundles
- The broadband L4 Redirect feature and the Intelligent Services Gateway feature
- Per-user firewall
- Lawful intercept
- MLP with MPLS-TE FRR
- Change of Authorization (CoA)
- Layer 2 input QoS classification
- The Multiclass Multilink Protocol (MCMP) RFC 2686 extension to LFI
- Per-user Access Control Lists (ACLs) applied through the RADIUS server are not supported. However, ACLs applied through the virtual template definition for the bundle are supported.
- Only the MLP long-sequence number format is supported for the packet header format option.

Additional References

The following sections provide references related to the Multilink ppp protocol connections.

Related Documents

Related Topic	Document Title
Cisco IOS commands	*Cisco IOS Master Commands List, All Releases*
Configuring Multilink PPP Connections for Broadband and Serial Topologies	*Configuring Multilink PPP Connections for Broadband and Serial Topologies*
MLP	*Wide-Area Networking Configuration Guide: Multilink PPP, Cisco IOS XE Release 3S*

Related Topic	Document Title
PPP commands	*Cisco IOS Dial Technologies Command Reference*
Broadband Configuration	*Cisco IOS XE Broadband and DSL Configuration Guide*
Cisco IOS Configuration Fundamentals	*Cisco IOS Configuration Fundamentals Command Reference*

Standards

Standard	Title
None	—

MIBs

MIB	MIBs Link
None	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use the Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

RFC[1]	Title
RFC 1990	The PPP Multilink Protocol (MP)
RFC 2686	The Multi-Class Extension to Multi-Link PPP

1. Not all the supported RFCs are listed.

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for Multilink PPP Support for Cisco ASR 1000 Series Routers

Table 8-2 lists the features in this module and provides links to specific configuration information. Only features that were introduced or modified in Cisco IOS XE Release 2.2.0S or a later release appear in the table.

Not all commands may be available in your Cisco IOS software release. For release information about a specific command, see the corresponding command reference documentation.

Use Cisco Feature Navigator to find information about platform support and software image support. The Cisco Feature Navigator enables you to determine which Cisco IOS and Cisco Catalyst operating system software images support a specific software release, feature set, or platform. To access the Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 8-2 Feature Information for Multilink PPP Support for Cisco ASR 1000 Series Routers

Feature Name	Releases	Feature Information
MLPoSerial	Cisco IOS XE Release 2.2.0S	In Cisco IOS XE Release 2.2.(0)S, support for MLPoSerial was introduced on the Cisco ASR 1000 Series Aggregation Services Routers.
MLPoBroadband with single-link bundles	Cisco IOS XE Release 3.4.0S	In Cisco IOS XE Release 3.4.(0)S, support for MLPoBroadband with single-link bundles was introduced on the Cisco ASR 1000 Series Aggregation Services Routers. Support for MLP upstream fragment reassembly was also added, but not for downstream fragmentation.
MLPoLNS and MLPoE/MLPoVLAN/MLPoQinQ with up to 8 links per bundle	Cisco IOS XE Release 3.7.1S	In Cisco IOS XE Release 3.7.1S, support for MLPoLNS, MLPoE, MLPoVLAN, and MLPoQinQ with up to eight links per bundle was introduced on the Cisco ASR 1000 Series Aggregation Services Routers. Support for downstream MLP LFI for all broadband MLPs was also added.

Table 8-2 Feature Information for Multilink PPP Support for Cisco ASR 1000 Series Routers

Feature Name	Releases	Feature Information
MLPoLNS Model F and IP Tunnel Marking	Cisco IOS XE Release 3.7.1S	In Cisco IOS XE Release 3.7.1S, support for MLPoLNS Model F and IP Tunnel Marking was introduced on the Cisco ASR 1000 Series Aggregation Services Routers. **Note** Model-F downstream shaping for MLPoLNS is not supported on the Cisco ASR 1002-X Router and the FP100 line card.
Multi member-link MLPPPoA/MLPPPoEoA (including DS LFI)	Cisco IOS XE Release 3.7.12S	In Cisco IOS XE Release 3.7.12S, support for Multi member-link MLPPPoA/MLPPPoEoA (including DS LFI) was introduced on the Cisco ASR 1000 Series Aggregation Services Routers.

Note For more information on the maximum scale numbers supported in various releases, see Table 8-1.

UniDirectional Link Detection (UDLD) Protocol

First Published: March 28, 2013

This document describes how to configure the UniDirectional Link Detection (UDLD) protocol on the Cisco ASR 1000 Series Aggregation Services Routers.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest information about features and caveats, see the release notes document pertaining to your platform and software release. To find information about the features documented in this module and to view a list of the releases in which each feature is supported, see the "Feature Information for Configuring UDLD on Cisco ASR 1000 Series Aggregation Services Routers" section on page 9-14.

Use the Cisco Feature Navigator to find information about platform support and Cisco IOS and Cisco Catalyst operating system software image support. To access the Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Contents

- Restrictions for the UDLD Protocol, page 9-1
- Information About the UDLD Protocol, page 9-2
- How to Configure the UDLD Protocol, page 9-3
- Configuration Examples for UDLD Protocol, page 9-10
- Additional References, page 9-12
- Feature Information for Configuring UDLD on Cisco ASR 1000 Series Aggregation Services Routers, page 9-14

Restrictions for the UDLD Protocol

Currently, the UDLD protocol on the Cisco ASR 1000 Series Aggregation Services Routers has the following limitations:

- High Availability (HA) is not supported, but when the Ethernet port is up and UDLD is enabled on the port, the UDLD automatically performs the detection.
- Only Gigabit Ethernet, 10 Gigabit Ethernet, and Fast Ethernet interfaces are supported.
- Supports only the basic UDLD functions.

Information About the UDLD Protocol

These sections describe how UDLD works:

- UDLD Overview, page 9-2
- Configuring the UDLD Aggressive Mode, page 9-3
- Default UDLD Configuration, page 9-3

UDLD Overview

The Cisco-proprietary UDLD protocol allows the devices connected through fiber optic or copper (for example, Category 5 cabling) Ethernet cables that are connected to the LAN ports to monitor the physical configuration of the cables and detect whether a unidirectional link exists. When a unidirectional link is detected, the UDLD shuts down the affected LAN port and alerts the corresponding user, because unidirectional links cause a variety of problems, including spanning tree topology loops.

UDLD is a Layer 2 protocol that works with the Layer 1 protocols to determine the physical status of a link. In Layer 1, auto negotiation takes care of physical signaling and fault detection. UDLD performs tasks that auto negotiation cannot perform, such as detecting the identities of neighbors and shutting down misconnected LAN ports. When you enable both auto negotiation and UDLD, the Layer 1 and Layer 2 detections work together to prevent physical and logical unidirectional connections and the malfunctioning of other protocols.

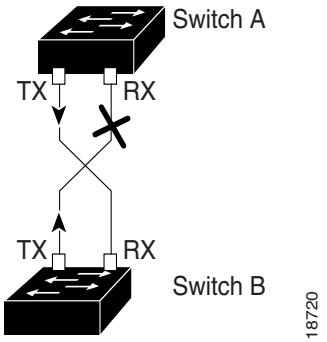
A unidirectional link occurs whenever the traffic transmitted by a local device over a link is received by a neighbor, but traffic transmitted from the neighbor is not received by the local device. If one of the fiber strands in a pair is disconnected, the link does not stay up as long as the auto negotiation is active. In such a scenario, the logical link is undetermined, and the UDLD does not take any action. If both the fibers are working normally in Layer 1, the UDLD in Layer 2 determines whether those fibers are connected correctly and whether the traffic is flowing bidirectionally between the correct neighbors. This check cannot be performed by auto negotiation because auto negotiation operates in Layer 1.

The Cisco ASR 1000 Series Aggregation Services Routers periodically transmit the UDLD packets to the neighbor devices on LAN ports where UDLD is enabled. If the packets are echoed back within a specific timeframe and they are lacking a specific acknowledgment (echo), the link is flagged as unidirectional and the LAN port is shut down. Devices on both ends of the link must support UDLD for the protocol to successfully identify and disable the unidirectional links.

Note By default, the UDLD is disabled on all ports to avoid sending unnecessary traffic.

Figure 9-1 shows an example of a unidirectional link condition. Switch B successfully receives traffic from Switch A on the port. However, Switch A does not receive traffic from Switch B on the same port. UDLD detects the problem and disables the port.

Figure 9-1 Unidirectional Link



Configuring the UDLD Aggressive Mode

Configure the UDLD aggressive mode only on the point-to-point link between the network devices that support the UDLD aggressive mode. With UDLD aggressive mode enabled, a port on a bidirectional link that has a UDLD neighbor relationship established stops receiving the UDLD packets. The UDLD tries to re-establish the connection with the neighbor; the port is disabled after eight failed retries.

To prevent spanning tree loops, nonaggressive UDLD with the default interval of 15 seconds is fast enough to shut down a unidirectional link before a blocking port transitions to the forwarding state (with default spanning tree parameters).

When the UDLD aggressive mode is enabled, the UDLD will error disable the ports on the link to prevent the traffic from being discarded under the following scenarios:

- One side of a link has a port (either Tx and Rx) stuck.
- One side of a link remains up while the other side of the link has gone down.

Default UDLD Configuration

Table 9-1 shows the default UDLD configuration.

Table 9-1 UDLD Default Configuration

Feature	Default Value
UDLD global enable state	Globally disabled
UDLD aggressive mode	Disabled
UDLD per-port enable state for fiber-optic media	Disabled
UDLD per-port enable state for twisted-pair (copper) media	Disabled on all Ethernet 10/100 and 1000BASE-TX LAN ports

How to Configure the UDLD Protocol

These sections describe how to configure the UDLD protocol:

- Enabling UDLD Globally, page 9-4

- Enabling UDLD on Individual LAN Interfaces, page 9-4
- Disabling UDLD on Fiber-Optic LAN Interfaces, page 9-6
- Configuring the UDLD Probe Message Interval, page 9-7
- Resetting the Disabled LAN Interfaces Manually, page 9-8
- Resetting the Disabled LAN Interfaces Automatically, page 9-8
- Debugging UDLD, page 9-9

Enabling UDLD Globally

To globally enable the UDLD on all fiber-optic LAN ports, perform this task:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **udld** {**enable** | **aggressive**}

DETAILED STEPS

	Command	Purpose
Step 1	**enable** **Example:** `Router# enable`	Enables the privileged EXEC mode. Enter your password, if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters the global configuration mode.
Step 3	**udld {enable | aggressive}** **no udld {enable | aggressive}** **Example:** `Router(config)# udld enable`	Enables the UDLD globally on fiber-optic LAN ports. **Note** This command configures only the fiber-optic LAN ports. Individual LAN port configuration overrides the setting of this command. Use the **no** form of this command to disable the UDLD globally on fiber-optic LAN ports.

Enabling UDLD on Individual LAN Interfaces

To enable the UDLD on individual LAN interfaces, perform this task:

SUMMARY STEPS

1. **enable**
2. **configure terminal**

3. **interface** *type slot/port*
4. **udld port** [**aggressive**]

DETAILED STEPS

	Command	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables the privileged EXEC mode. Enter your password, if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters the global configuration mode.
Step 3	**interface** *type slot/port* **Example:** `Router(config)# interface gigabitethernet2/2`	Selects the LAN port to configure.
Step 4	**udld port [aggressive]** **no udld port [aggressive]** **Example:** `Router(config)# udld port aggressive`	Enables UDLD on a specific LAN port. Enter the **aggressive** keyword to enable the aggressive mode. On a fiber-optic LAN port, this command overrides the **udld enable** global configuration command setting. Use the **no** form of this command to disable the UDLD on a nonfiber-optic LAN port. On fiber-optic LAN ports, the **no udld port** command reverts the LAN port configuration to the **udld enable** global configuration command setting.

Disabling UDLD on Fiber-Optic LAN Interfaces

To disable the UDLD on individual fiber-optic LAN ports, perform this task:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type slot/port*
4. **udld port disable**

DETAILED STEPS

	Command	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables the privileged EXEC mode. Enter your password, if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters the global configuration mode.
Step 3	**interface** *type slot/port* **Example:** `Router(config)# interface gigabitethernet2/2`	Selects the LAN port to configure.
Step 4	**udld port disable** **no udld port disable** **Example:** `Router(config)# udld port disable`	Disables UDLD on a fiber-optic LAN port. Use the **no** form of this command to revert to the **udld enable** global configuration command setting. **Note** This command is supported only on the fiber-optic LAN ports.

Configuring the UDLD Probe Message Interval

To configure the time between UDLD probe messages on ports that are in the advertisement mode and are currently determined to be bidirectional, perform this task:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **udld message time** *interval*

DETAILED STEPS

	Command	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables the privileged EXEC mode. Enter your password, if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters the global configuration mode.
Step 3	**udld message time** *interval* **no udld message** **Example:** `Router(config)# udld message time 60`	Configures the time between the UDLD probe messages on the ports that are in the advertisement mode and are currently determined to be bidirectional. Valid values are from 7 to 90 seconds. Use the **no** form of this command to return to the default value (15 seconds).

Resetting the Disabled LAN Interfaces Manually

To reset all the LAN ports that have been shut down by UDLD, perform this task:

SUMMARY STEPS

1. **enable**
2. **udld reset**

DETAILED STEPS

	Command	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables the privileged EXEC mode. Enter your password, if prompted.
Step 2	**udld reset** **Example:** `Router# udld reset`	Resets all the LAN ports that have been shut down by UDLD.

Resetting the Disabled LAN Interfaces Automatically

To automatically reset all the LAN ports that have been shut down by UDLD, perform this task:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **udld recovery**
4. **udld recovery interval** *interval*

DETAILED STEPS

	Command	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables the privileged EXEC mode. Enter your password, if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters the global configuration mode.

	Command	Purpose
Step 3	**udld recovery** **no udld recovery** **Example:** `Router(config)# udld recovery`	Enables the recovery timer for the UDLD error disabled state. Use the **no** form of this command to disable the recovery timer for the UDLD error disabled state.
Step 4	**udld recovery interval** *interval* **no udld recovery interval** **Example:** `Router(config)# udld recovery interval 100`	Specifies the time to recover from a UDLD error disabled state. Valid values are from 30 to 86400 seconds. Use the **no** form of this command to return to the default value (300 seconds).

Debugging UDLD

To enable the debugging of an UDLD activity, perform this task:

SUMMARY STEPS

1. **enable**
2. **debug udld {events | packets | registries}**

DETAILED STEPS

	Command	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables the privileged EXEC mode. Enter your password, if prompted.
Step 2	**debug udld {events | packets | registries}** **no debug udld {events | packets | registries}** **Example:** `Router# debug udld events`	Enables the debugging of UDLD process events, packets, or registry events. Use the **no** form of this command to disable the debugging of UDLD process events, packets, or registry events.

Configuration Examples for UDLD Protocol

The secion provides the following configuration examples:

Example: Verifying a UDLD Configuration, page 9-10
Example: Verifying Information About Neighbors, page 9-10
Example: Displaying all the UDLD Interface Statuses, page 9-11

Example: Verifying a UDLD Configuration

The following example show how to use the **show** command to verify an UDLD configuration:

Sample Output for the show udld interface-id Command

```
Router# show udld gigabitethernet2/2

Interface Gi2/2
---

Port enable administrative configuration setting: Follows device default
Port enable operational state: Enabled
Current bidirectional state: Bidirectional
Current operational state: Advertisement
Message interval: 60
Time out interval: 5
No multiple neighbors detected

    Entry 1
---
    Expiration time: 146
    Device ID: 1
    Current neighbor state: Bidirectional
    Device name: 0050e2826000
    Port ID: 2/1
    Neighbor echo 1 device: SAD03160954
    Neighbor echo 1 port: Gi1/1
    Message interval: 5
    CDP Device name: 066527791
```

Example: Verifying Information About Neighbors

The following example shows how to view the information pertaining to neighbors:

Sample Output for the show udld neighbors Command

```
Router# show udld neighbors

Port     Device Name                    Device ID    Port-ID OperState

-------- ------------------------------ ------------ ------- --------------

Gi3/1    SAL0734K5R2                    1            Gi4/1   Bidirectional

Gi4/1    SAL0734K5R2                    1            Gi3/1   Bidirectional
```

Example: Displaying all the UDLD Interface Statuses

The follwing example shows how to display all the UDLD interface statuses:

Sample Output for the show udld Command

```
Router# show udld

Interface Gi0/0/0
---
Port enable administrative configuration setting: Follows device default
Port enable operational state: Disabled
Current bidirectional state: Unknown

Interface Gi0/0/1
---
Port enable administrative configuration setting: Follows device default
Port enable operational state: Disabled
Current bidirectional state: Unknown

Interface Fa0/1/0
---
Port enable administrative configuration setting: Disabled
Port enable operational state: Disabled
Current bidirectional state: Unknown

Interface Fa0/1/1
---
Port enable administrative configuration setting: Disabled
Port enable operational state: Disabled
.
.
.
```

Additional References

The following sections provide references related to the UniDirectional Link Detection (UDLD) protocol on the Cisco ASR 1000 Series Aggregation Services Routers.

Related Documents

Related Topic	Document Title
Cisco IOS commands	*Cisco IOS Master Commands List, All Releases*
Cisco IOS Configuration Fundamentals	*Cisco IOS Configuration Fundamentals Command Reference*

Standards

Standard	Title
No new or modified standards are supported by this feature.	—

MIBs

MIB	MIBs Link
None	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use the Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

RFC[1]	Title
RFC 5171	Cisco Systems UniDirectional Link Detection (UDLD) Protocol

1. Not all the supported RFCs are listed.

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for Configuring UDLD on Cisco ASR 1000 Series Aggregation Services Routers

Table 9-2 lists the features in this module and provides links to specific configuration information. Only features that were introduced or modified in Cisco IOS Release 3.9S or later appear in the table.

Not all commands may be available in your Cisco IOS software release. For release information about a specific command, see the corresponding command reference documentation.

Use the Cisco Feature Navigator to find information about platform support and software image support. The Cisco Feature Navigator enables you to determine which Cisco IOS and Cisco Catalyst operating system software images support a specific software release, feature set, or platform. To access the Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 9-2 Feature Information for Configuring UDLD on Cisco ASR 1000 Series Aggregation Services Routers

Feature Name	Releases	Feature Information
UniDirectional Link Detection (UDLD) protocol	3.9S	The Cisco-proprietary UDLD protocol allows devices connected through fiber-optic or copper (for example, Category 5 cabling) Ethernet cables connected to LAN ports to monitor the physical configuration of the cables and detect when a unidirectional link exists. When a unidirectional link is detected, UDLD shuts down the affected LAN port and alerts users. Unidirectional links can cause a variety of problems, including spanning tree topology loops. In Cisco IOS XE Release 3.9S, this feature was implemented on the Cisco ASR 1000 Series Aggregation Services Routers. The following sections provide information about this feature: • Information About the UDLD Protocol, page 9-2 • How to Configure the UDLD Protocol, page 9-3

Using the Management Ethernet Interface

The Cisco ASR 1000 Series Routers have one Gigabit Ethernet Management Ethernet interface on each Route Processor.

Finding Feature Information in This Module

Your software release might not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the "Feature Information for Using the Management Ethernet Interface" section on page 10-8.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Contents

This guide covers the following topics:

- Gigabit Ethernet Management Interface Overview, page 10-1
- Gigabit Ethernet Port Numbering, page 10-2
- Gigabit Ethernet Port Numbering, page 10-2
- IP Address Handling in ROMmon and the Management Ethernet Port, page 10-2
- Gigabit Ethernet Management Interface VRF, page 10-3
- Common Ethernet Management Tasks, page 10-3

Gigabit Ethernet Management Interface Overview

The purpose of this interface is to allow users to perform management tasks on the router; it is basically an interface that should not and often cannot forward network traffic but can otherwise access the router, often via Telnet and SSH, and perform most management tasks on the router. The interface is most useful before a router has begun routing, or in troubleshooting scenarios when the SPA interfaces are inactive.

The following aspects of the Management Ethernet interface should be noted:

- Each RP has a Management Ethernet interface, but only the active RP has an accessible Management Ethernet interface (the standby RP can be accessed using the console port, however).
- IPv4, IPv6, and ARP are the only routed protocols supported for the interface.
- The interface provides a method of access to the router even if the SPA interfaces or the IOS processes are down.
- The Ethernet Management Interface cannot be used as a Lawful Intercept MD source interface.
- The Management Ethernet interface is part of its own VRF. This is discussed in more detail in the "Gigabit Ethernet Management Interface VRF" section on page 10-3.

Gigabit Ethernet Port Numbering

The Gigabit Ethernet Management port is always GigabitEthernet0.

In a dual RP configuration, the Management Ethernet interface on the active RP will always be Gigabit Ethernet 0, while the Management Ethernet interface on the standby RP will not be accessible using the Cisco IOS CLI in the same telnet session. The standby RP can be telnetted to through the console port, however.

The port can be accessed in configuration mode like any other port on the Cisco ASR 1000 Series Routers:

```
Router#config t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#interface gigabitethernet0
Router(config-if)#
```

IP Address Handling in ROMmon and the Management Ethernet Port

On the Cisco ASR 1000 Series Routers, IP addresses can be configured in ROMmon (the **IP_ADDRESS=** and **IP_SUBNET_MASK=** commands) and through the use of the IOS command-line interface (the **ip address** command in interface configuration mode).

Assuming the IOS process has not begun running on the Cisco ASR 1000 Series Router, the IP address that was set in ROMmon acts as the IP address of the Management Ethernet interface. In cases where the IOS process is running and has taken control of the Management Ethernet interface, the IP address specified when configuring the Gigabit Ethernet 0 interface in the IOS CLI becomes the IP address of the Management Ethernet interface. The ROMmon-defined IP address is only used as the interface address when the IOS process is inactive.

For this reason, the IP addresses specified in ROMmon and in the IOS CLI can be identical and the Management Ethernet interface will function properly in single RP configurations.

In dual RP configurations, however, users should never configure the IP address in the ROMmon on either RP0 or RP1 to match each other or the IP address as defined by the IOS CLI. Configuring matching IP addresses introduces the possibility for an active and standby Management Ethernet interface having the same IP address with different MAC addresses, which will lead to unpredictable traffic treatment.

Gigabit Ethernet Management Interface VRF

The Gigabit Ethernet Management interface is automatically part of its own VRF. This VRF, which is named "Mgmt-intf," is automatically configured on the Cisco ASR 1000 Series Router and is dedicated to the Management Ethernet interface; no other interfaces can join this VRF. Therefore, this VRF does not participate in the MPLS VPN VRF or any other network-wide VRF.

Placing the management ethernet interface in its own VRF has the following effects on the Management Ethernet interface:

- Many features must be configured or used inside the VRF, so the CLI may be different for certain Management Ethernet functions on the Cisco ASR 1000 Series Routers than on Management Ethernet interfaces on other routers.
- Prevents transit traffic from traversing the router. Because all of the SPA interfaces and the Management Ethernet interface are automatically in different VRFs, no transit traffic can enter the Management Ethernet interface and leave a SPA interface, or vice versa.
- Improved security of the interface. Because the Mgmt-intf VRF has its own routing table as a result of being in its own VRF, routes can only be added to the routing table of the Management Ethernet interface if explicitly entered by a user.

The Management Ethernet interface VRF supports both IPv4 and IPv6 address families.

Common Ethernet Management Tasks

Because users can perform most tasks on a router through the Management Ethernet interface, many tasks can be done by accessing the router through the Management Ethernet interface.

This section documents tasks that might be common or slightly tricky on the Cisco ASR 1000 Series Routers. It is not intended as a comprehensive list of all tasks that can be done using the Management Ethernet interface.

This section covers the following processes:

- Viewing the VRF Configuration, page 10-4
- Viewing Detailed VRF Information for the Management Ethernet VRF, page 10-4
- Setting a Default Route in the Management Ethernet Interface VRF, page 10-4
- Setting the Management Ethernet IP Address, page 10-4
- Telnetting over the Management Ethernet Interface, page 10-5
- Pinging over the Management Ethernet Interface, page 10-5
- Copy Using TFTP or FTP, page 10-5
- NTP Server, page 10-6
- SYSLOG Server, page 10-6
- SNMP-Related Services, page 10-6
- Domain Name Assignment, page 10-6
- DNS service, page 10-6
- RADIUS or TACACS+ Server, page 10-6
- VTY lines with ACL, page 10-7

Viewing the VRF Configuration

The VRF configuration for the Management Ethernet interface is viewable using the **show running-config vrf** command.

This example shows the default VRF configuration:

```
Router# show running-config vrf
Building configuration...

Current configuration : 351 bytes
vrf definition Mgmt-intf
 !
 address-family ipv4
 exit-address-family
 !
 address-family ipv6
 exit-address-family
!
(some output removed for brevity)
```

Viewing Detailed VRF Information for the Management Ethernet VRF

To see detailed information about the Management Ethernet VRF, enter the **show vrf detail Mgmt-intf** command:

```
Router# show vrf detail Mgmt-intf
VRF Mgmt-intf (VRF Id = 4085); default RD <not set>; default VPNID <not set>
  Interfaces:
    Gi0
Address family ipv4 (Table ID = 4085 (0xFF5)):
  No Export VPN route-target communities
  No Import VPN route-target communities
  No import route-map
  No export route-map
  VRF label distribution protocol: not configured
  VRF label allocation mode: per-prefix
Address family ipv6 (Table ID = 503316481 (0x1E000001)):
  No Export VPN route-target communities
  No Import VPN route-target communities
  No import route-map
  No export route-map
  VRF label distribution protocol: not configured
  VRF label allocation mode: per-prefix
```

Setting a Default Route in the Management Ethernet Interface VRF

To set a default route in the Management Ethernet Interface VRF, enter the following command

ip route vrf Mgmt-intf 0.0.0.0 0.0.0.0 *next-hop-IP-address*

Setting the Management Ethernet IP Address

The IP address of the Management Ethernet port is set like the IP address on any other interface.

Below are two simple examples of configuring an IPv4 adress and an IPv6 address on the Management Ethernet interface.

IPv4 Example

```
Router(config)# interface GigabitEthernet 0
Router(config-if)# ip address A.B.C.D A.B.C.D
```

IPv6 Example

```
Router(config)# interface GigabitEthernet 0
Router(config-if)# ipv6 address X:X:X:X::X
```

Telnetting over the Management Ethernet Interface

Telnetting can be done through the VRF using the Management Ethernet interface.

In the following example, the router telnets to 172.17.1.1 through the Management Ethernet interface VRF:

```
Router# telnet 172.17.1.1 /vrf Mgmt-intf
```

Pinging over the Management Ethernet Interface

Pinging other interfaces using the Management Ethernet interface is done through the VRF.

In the following example, the router pings the interface with the IP address of 172.17.1.1 through the Management Ethernet interface:

```
Router# ping vrf Mgmt-intf 172.17.1.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.17.1.1, timeout is 2 seconds:
.!!!!
Success rate is 80 percent (4/5), round-trip min/avg/max = 1/1/1 ms
```

Copy Using TFTP or FTP

To copy a file using TFTP through the Management Ethernet interface, the **ip tftp source-interface GigabitEthernet 0** command must be entered before entering the **copy tftp** command because the **copy tftp** command has no option of specifying a VRF name.

Similarly, to copy a file using FTP through the Management Ethernet interface, the **ip ftp source-interface GigabitEthernet 0** command must be entered before entering the **copy ftp** command because the **copy ftp** command has no option of specifying a VRF name.

TFTP Example

```
Router(config)# ip tftp source-interface gigabitethernet 0
```

FTP Example

```
Router(config)# ip ftp source-interface gigabitethernet 0
```

NTP Server

To allow the software clock to be synchronized by a Network Time Protocol (NTP) time server over the Management Ethernet interface, enter the **ntp server vrf Mgmt-intf** command and specify the IP address of the device providing the update.

The following CLI provides an example of this procedure.

```
Router(config)# ntp server vrf Mgmt-intf 172.17.1.1
```

SYSLOG Server

To specify the Management Ethernet interface as the source IP or IPv6 address for logging purposes, enter the **logging host <ip-address> vrf Mgmt-intf** command.

The following CLI provides an example of this procedure.

```
Router(config)# logging host <ip-address> vrf Mgmt-intf
```

SNMP-Related Services

To specify the Management Ethernet interface as the source of all SNMP trap messages, enter the **snmp-server source-interface traps gigabitEthernet 0** command.

The following CLI provides an example of this procedure:

```
Router(config)# snmp-server source-interface traps gigabitEthernet 0
```

Domain Name Assignment

The IP domain name assignment for the Management Ethernet interface is done through the VRF.

To define the default domain name as the Management Ethernet VRF interface, enter the **ip domain-name vrf Mgmt-intf** *domain* command.

```
Router(config)# ip domain-name vrf Mgmt-intf cisco.com
```

DNS service

To specify the Management Ethernet interface VRF as a name server, enter the **ip name-server vrf Mgmt-intf** *IPv4-or-IPv6-address* command.

```
Router(config)# ip name-server vrf Mgmt-intf IPv4-or-IPv6-address
```

RADIUS or TACACS+ Server

To group the Management VRF as part of a AAA server group, enter the **ip vrf forward Mgmt-intf** command when configuring the AAA server group.

The same concept is true for configuring a TACACS+ server group. To group the Management VRF as part of a TACACS+ server group, enter the **ip vrf forwarding Mgmt-intf** command when configuring the TACACS+ server group.

RADIUS Server Group Configuration

```
Router(config)# aaa group server radius hello
Router(config-sg-radius)# ip vrf forwarding Mgmt-intf
```

TACACS+ Server Group Example

```
outer(config)# aaa group server tacacs+ hello
Router(config-sg-tacacs+)# ip vrf forwarding Mgmt-intf
```

VTY lines with ACL

To ensure an access control list (ACL) is attached to vty lines that are and are not using VRF, use the **vrf-also** option when attaching the ACL to the vty lines.

```
Router(config)# line vty 0 4
Router(config-line)# access-class 90 in vrf-also
```

Additional References

Standards

Standard	Title
None	—

MIBs

MIB	MIBs Link
None	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at this URL: http://www.cisco.com/go/mibs

RFCs

RFC	Title
None	—

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for Using the Management Ethernet Interface

Table 10-1 lists the features in this module and provides links to specific configuration information.

Use Cisco Feature Navigator to find information about platform support and software image support. Cisco Feature Navigator enables you to determine which software images support a specific software release, feature set, or platform. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Note Table 10-1 lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Table 10-1 Feature Information for Using the Management Ethernet Interface

Feature Name	Releases	Feature Information
Using the Management Ethernet Interface	Cisco IOS XE 2.1S	This feature was introduced.

Network Synchronization Support

Most of the services that are provided over networks must be fully synchronized with one another in order to operate efficiently. If the network devices that constitute a network do not operate at the same clock rates, there is an overall decrease in the performance of the network and a consequent degradation in the quality of the services offered by the network. This document explains how to configure network synchronization on the Cisco ASR 1000 Series Aggregation Services Routers.

Finding Feature Information

Your software release might not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the "Feature Information for Network Synchronization Support" section on page 11-24.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Contents

This guide covers the following topics:

- Prerequisites for Network Synchronization Support, page 11-2
- Network Synchronization Overview, page 11-2
- Synchronization Status Message and Ethernet Synchronization Messaging Channel, page 11-6
- Restrictions for Network Synchronization, page 11-7
- Configuring Network Synchronization, page 11-8

Prerequisites for Network Synchronization Support

The following goals must be taken into account while designing the synchronization plan for a network:

- Synchronize the greatest number of network elements to the smallest number of independent clock sources. Ideally, all the network elements should be synchronized to a single clock source.
- Use clock sources of the highest quality (in terms of stability and long-term accuracy).
- To ensure resiliency of synchronization, plan for possible failure of the clock sources, network elements, and network trunks.

Network Synchronization Overview

Synchronous Ethernet (SyncE), which is defined by the Telecommunication Standardization Sector (ITU-T) standards, such as G.8261 and G.8262, leverages the PHY layer of the Ethernet to transmit clock information to remote sites.

SyncE provides a cost-effective alternative to the Synchronous Optical Networking (SONET) networks. For SyncE to work, each network element along the synchronization path must support SyncE. To implement SyncE, the bit clock of the Ethernet is aligned to a reliable clock that is traceable to the Primary Reference Clock (PRC).

Features Supported on ASR 1000 Platforms

Platform Support	Features Supported	Features Not Supported
ASR 1001-X	Synchronous Ethernet Feature is supported with Built-in Gigabit and 10-Gigabit Ethernet Ports	NetClk Feature
ASR1002-X	Synchronous Ethernet Feature is supported with Built-in Gigabit Ethernet Ports.	NetClk Feature
ASR1001	—	NetClk and Network Synchronization Feature
ASR 1002, ASR 1004, ASR 1006 and ASR 1013	NetClk is supported on all ASR1000 family routers apart from ASR1001, ASR1002-X, and ASR 1001-X.	—

Note Network-Clocking feature is not supported with SFP-GE-T on any of the ASR1000 platforms.

Cisco ASR 1000 Series Aggregation Services Routers have a dedicated external interface known as BITS interface to recover clocking information from a Synchronization Supply Unit (SSU). They use this clock for SyncE. The BITS interface supports E1 (European SSUs) and T1 (American BITS) framing.

Note The Cisco ASR 1001-X Router does not have an on-board BITS port, hence it does not support those features that use BITS ports.

Table 11-1 lists the framing modes for a BITS port on the Cisco ASR 1002-X Router.

Table 11-1 Framing Modes for a BITS or SSU Port on the Cisco ASR 1002-X Router

BITS or SSU Port Support Matrix	Framing Modes Supported	SSM or QL Support	Tx Port	Rx Port
T1	T1 ESF	Yes	Yes	Yes
T1	T1 SF	Yes	Yes	Yes
T1	T1 D4	Yes	Yes	Yes
E1	E1 CRC4	Yes	Yes	Yes
E1	E1 FAS	Yes	Yes	Yes
E1	E1 CAS	Yes	Yes	Yes
E1	E1 CAS CRC4	Yes	Yes	Yes
2048 kHz	2048 kHz	Yes	Yes	Yes

Table 11-2 lists the framing modes for a BITS port on the other Cisco ASR 1000 Series Aggregation Services Routers.

Table 11-2 Framing Modes for a BITS or SSU Port on the Other Cisco ASR 1000 Series Aggregation Services Routers

BITS or SSU Port Support Matrix	Framing Modes Supported	SSM or QL Support	Tx Port	Rx Port
T1	T1 ESF	No	No	Yes
T1	T1 SF	No	No	Yes
E1	E1 CRC4	No	No	Yes
E1	E1 FAS	No	No	Yes
E1	E1 CAS	No	No	Yes
E1	E1 CAS CRC4	No	No	Yes
2048 kHz	2048 kHz	No	No	Yes

You can configure network synchronization on the Cisco ASR 1000 Series Aggregation Services Routers by using one of the following features:

- Clock Recovery from SyncE

  When this feature is configured, the system clock is recovered from the SyncE clocking source (SyncE-enabled interfaces only). The router uses this clock as the Tx clock for other supported Shared Port Adapters (SPAs).

- Clock Recovery from an External Interface

  When this feature is enabled, the system clock is recovered from a BITS clock source.

- Clock Recovery from a GPS (10M) Source

Note This feature is supported only on the Cisco ASR 1002-X Router.

When this feature is enabled, the system clock is recovered from a GPS (10M) clock source.

- Line to External

Note This feature is not supported on the Cisco ASR 1002-X Router.

When this feature is enabled, the clock received from an Ethernet is forwarded to an external SSU. The SyncE feature provides the clock cleanup functionality. When a router is in the middle of a synchronization chain, the received clock may experience unacceptable wander and jitter. The router recovers the clock from the SyncE interface, converts it to the format required for the BITS interface, and sends it to an SSU through the BITS port. The SSU then performs a cleanup and sends it back to the BITS interface. The cleaned up clock is received from the SSU and is used as the Tx clock for the SyncE ports. In the context of the Cisco ASR 1000 Series Aggregation Services Routers, the interface from which the clock is recovered and the BITS port to the SSU must reside on the same card.

A SyncE-enabled Cisco ASR 1000 Series Router provides the Squelching feature. In this feature, an alarm indication signal (AIS) is sent to the Tx interfaces if the clock source goes down. When the Line to External feature is configured, if the line source goes down at any point of time, an AIS is transmitted through the external interface to the SSU. Squelching can be performed only on external devices, such as SSUs and PRCs.

You can have a maximum of eight clock sources configured on a Cisco ASR 1000 Series Router. On a router that supports the Network Synchronization feature, you can configure selection of the clock source on the basis of the quality of the clock source. With this feature configured, the clock source that offers the best quality is made the default clock source. If the highest level of quality is offered by more than one clock source, the clock source with the highest priority is made the default clock source. You can also manage synchronization by using the following management options:

- Hold-Off Time: If a clock source goes down, the router waits for a specific hold-off time before removing the clock source from the clock selection process. By default, the value of hold-off time is 300 milliseconds (ms).
- Wait to Restore: If a SyncE interface comes up, the router waits for a specific period of time before considering the SyncE interface for synchronization source. By default, the value is 300 seconds.
- Force Switch: Forcefully selects a synchronization source irrespective of whether the source is available or within the specified range.
- Manual Switch: Manually select a synchronization source, provided the clock source has a higher quality level than the currently active clock.

Table 11-3 lists the SPAs that support network synchronization.

Table 11-3 SPAs supporting Network Synchronization

SPA Type	SPA Name
ASR 1001-X Built-in SPA	BUILT-IN-2T+6X1GE
ASR1002-X Built-in SPA	6XGE-BUILT-IN

Table 11-3 SPAs supporting Network Synchronization (continued)

SPA Type	SPA Name
Synchronous Ethernet	SPA-2XGE-SYNCE
Ethernet Line Card	ASR1000-2T+20X1GE
Serial and Channelized SPA	SPA-1XCHSTM1/OC3
	SPA-1XCHOC12/DS0
	SPA-8XCHT1/E1-V2 (Supported on the Cisco ASR 1002-X Router and the Cisco ASR1001-X Router, from Cisco IOS XE Release 3.11 onwards)
CEOP	SPA-1CHOC3-CE-ATM
	SPA-24CHT1-CE-ATM
ATM	SPA-1XOC3-ATM-V2
	SPA-3XOC3-ATM-V2
	SPA-1XOC12-ATM-V2
	SPA-2CHT3-CE-ATM
POS	SPA-2XOC3-POS
	SPA-4XOC3-POS
	SPA-4XOC3-POS-V2
	SPA-8XOC3-POS
	SPA-1XOC12-POS
	SPA-2XOC12-POS
	SPA-4XOC12-POS
	SPA-8XOC12-POS
	SPA-1XOC48-POS/RPR
	SPA-2XOC48POS/RPR
	SPA-4XOC48POS/RPR
	SPA-OC192POS-XFP

Note For more information on SPA support matrix see, http://www.cisco.com/en/US/docs/interfaces_modules/shared_port_adapters/configuration/ASR1000/ASRspasw.html

Note Netsync feature on SPA-2XGE-SYNCE is not supported on the Cisco ASR 1001-X Router.

Synchronization Status Message and Ethernet Synchronization Messaging Channel

Network clocking uses the following mechanisms to exchange the quality level of the clock between the network elements:

- Synchronization Status Message, page 11-6
- Ethernet Synchronization Messaging Channel, page 11-6

Synchronization Status Message

Network elements use Synchronization Status Messages (SSM) to inform the neighboring elements about the Quality Level (QL) of the clock. Non-Ethernet interfaces, such as optical interfaces and T1 or E1 SPA frames, use SSM. The key benefits of SSMs are:

- Prevents timing loops.Provides fast recovery when a part of the network fails.
- Ensures that a node gets timing from the most reliable clock source.

Ethernet Synchronization Messaging Channel

In order to maintain a logical communication channel in synchronous network connections, Ethernet relies on a channel called the Ethernet Synchronization Messaging Channel (ESMC), which is based on the IEEE 802.3 Organization-Specific Slow Protocol (OSSP) standards. ESMC relays the SSM code that represents the quality level of the Ethernet Equipment Clock (EEC) in a physical layer.

The ESMC packets are received only for the ports configured as clock sources, and transmitted on all the SyncE interfaces in the system. These packets are then processed by the clock selection algorithm on route processors (RP) and are used to select the best clock. The Tx frame is generated based on the QL value of the selected clock source, and sent to all the enabled SyncE ports.

Clock Selection Algorithm

The clock selection algorithm selects the best available synchronization source from the nominated sources. This algorithm exhibits nonrevertive behavior among the clock sources with the same QL value, and always selects the signal with the best QL value. For clock option 1, the default is revertive, and for clock option 2, the default is nonrevertive.

The clock selection process works in the following modes:

- QL-Enabled Mode, page 11-6
- QL-Disabled Mode, page 11-7

When multiple selection processes are present in a network element, all the processes work in the same mode.

QL-Enabled Mode

In QL-enabled mode, the following parameters contribute to the selection process:

- Quality level
- Signal fail via QL-FAILED

- Priority
- External commands.

If no external commands are active, the algorithm selects the reference (for clock selection) with the highest QL that does not experience a signal fail condition. If multiple inputs have the same highest QL, the input with the highest priority is selected. For multiple inputs having the same highest priority and QL, the existing reference is maintained (if it belongs to the highest priority and QL group). Otherwise, an arbitrary reference from the highest priority and QL group is selected.

QL-Disabled Mode

In QL-disabled mode, the following parameters contribute to the selection process:

- Signal failure
- Priority
- External commands

If no external commands are active, the algorithm selects the reference (for clock selection) with the highest priority that does not experience a signal fail condition. For multiple inputs having the same highest priority, the existing reference is maintained (if it belongs to highest priority group). Otherwise, an arbitrary reference from highest priority group is selected.

Restrictions for Network Synchronization

This section lists the restrictions for configuring network synchronization on a router. See Table 11-3 for a listing of the SPAs that you can use when configuring network synchronization.

- If the network clock algorithm is enabled, use the system clock as the Tx clock (synchronous mode) for the Ethernet interfaces of the corresponding router. You cannot change the synchronous mode on a per-interface basis.
- You can configure up to eight ports as clock sources on a router.
- The SyncE feature coexists with SSO, but is not SSO-compliant. The clock selection algorithm is restarted on a switchover. During switchover, the router goes into holdover mode.
- The SyncE interfaces in the WAN mode cannot be used for QL-enabled clock selection. You must either use them with the system in QL-disabled mode, or disable the ESMC on the interfaces, and use them as QL-disabled interfaces.
- We recommend that you do not configure multiple input sources with the same priority because this impacts the T_{SM} (switching message delay).
- You cannot implement the network clock-based clock selection algorithm and the new algorithm simultaneously. Both these are mutually exclusive.
- The Line to External feature for clock cleanup is supported only if the line interface and the external interface are on the same metronome SPA.

Note The Line to External feature is not supported on the Cisco ASR 1002-X Router.

Configuring Network Synchronization

You can configure network synchronization on the Cisco ASR 1000 Series Aggregation Services Routers by performing one of the following procedures:

- Configuring Clock Recovery from SyncE, page 11-8
- Configuring Clock Recovery from a BITS Port, page 11-9
- Configuring SyncE by Using the Line to External Feature, page 11-14

Configuring Clock Recovery from SyncE

This section describes how to configure clock recovery by using the SyncE method.

SUMMARY STEPS

1. **configure terminal**
2. **network-clock synchronization automatic**
3. **network-clock synchronization ssm option {1 | 2 {GEN1 | GEN2}}**
4. **interface gigabitethernet** *slot/card/port*
5. **synchronous mode**
6. **exit**
7. **network-clock input-source** *priority* **{interface** *interface-name slot/card/port* **| {external** *slot/card/port*}}
8. **exit**

DETAILED STEPS

	Command	Purpose

Step 1	`configure terminal`
	
	Example:
	`Router# configure terminal`
		Enters the global configuration mode.

Step 2	`network-clock synchronization automatic`
	
	Example:
	`Router(config)# network-clock synchronization automatic`
		Enables the network clock selection algorithm. This command disables the Cisco-specific network clock process, and turns on the G.781-based automatic clock selection process.

Step 3	`network-clock synchronization ssm option {1 | 2 {GEN1 | GEN2}}`
	
	Example:
	`Router(config)# network-clock synchronization ssm option 2 GEN1`
		Configures the equipment to work in a synchronization network. The *option_id* value 1 refers to a synchronization network design for Europe. This is the default value. The *option_id* value 2 refers to a synchronization network design for the U.S.

	Command	Purpose
Step 4	**interface gigabitethernet** *slot/card/port* **Example:** `Router(config)# interface gigabitethernet 0/2/0`	Specifies the Gigabit Ethernet interface to be configured: *slot/card/port*—Specifies the location of the interface.
Step 5	**synchronous mode** **Example:** `Router(config-if)# synchronous mode`	Sets the mode to synchronous mode.
Step 6	**exit** **Example:** `Router(config)# exit`	Exits the global configuration mode.

Step 7	**network-clock input-source priority {interface** *interface-name slot/card/port* **| {external** *slot/card/port***}}**
	
	`Router(config)# network-clock input-source 1 interface gigabitethernet 0/2/0`
	
	`Router(config)# network-clock input-source 1 external R0 2048k`
		Enables clock recovery from SyncE.
		
		The Cisco ASR1006 router supports two RP ports, one active and the other standby.

Step 8	**exit** **Example:** `Router(config)# exit`	Exits the global configuration mode.

Examples

The following example shows how to configure clock recovery from SyncE:

```
Router# configure terminal
Router(config)# network-clock synchronization automatic
Router(config)# network-clock synchronization ssm option 2 GEN1
Router(config)# interface gigabitethernet 0/2/0
Router(config-if)# synchronous mode
Router(config)# exit
Router(config)# network-clock input-source 1 interface gigabitethernet 0/2/0
Router(config)# exit
```

Configuring Clock Recovery from a BITS Port

This section describes how to configure clock recovery from a BITS port. BITS port can be configured either as an input-source or an output-source.

- Configuring Clock Recovery with a BITS Port as an Input-Source, page 11-10
- Configuring Clock Recovery with a BITS Port as an Output-Source, page 11-11

Configuring Clock Recovery with a BITS Port as an Input-Source

This section describes how to configure clock recovery with a BITS port as an input-source.

SUMMARY STEPS

1. **configure terminal**
2. **network-clock synchronization automatic**
3. **network-clock synchronization ssm option** *option-id generation-id*
4. **network-clock input-source** *priority* **{external** *slot/card/port***}**
5. **exit**

DETAILED STEPS

	Command	Purpose
Step 1	**enable** **Example:** `Router# enable`	Enables the privileged EXEC mode. Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters the global configuration mode.
Step 3	**network-clock synchronization automatic** **Example:** `Router(config)# network-clock synchronization automatic`	Enables the network clock selection algorithm. This command disables the Cisco-specific network clock process, and turns on the G.781-based automatic clock selection process.
Step 4	**network-clock synchronization ssm option {1 | 2 {GEN1 | GEN2}}** **Example:** `Router(config)# network-clock synchronization ssm option 2 GEN1`	Configures the equipment to work in the synchronization network. The *option_id* value 1 refers to synchronization network design for Europe. This is the default value. The *option_id* value 2 refers to the synchronization network design for the U.S.

	Command	Purpose

Step 5	**network-clock input-source priority {external** *slot/card/port***}**
	
	Example:
	`Router(config)# network-clock input-source 1 External 0/3/0 t1 esf`
	
	`Router(config)# network-clock input-source 1 External R0 t1 esf`
	
	Example:For ASR 1002-X Router:
	`Router(config)# network-clock input-source 1 external r0 e1 cas crc4 120 linecode`
		Enables clock recovery from a BITS port.
		
		Note Routers with dual hardware support, such as the Cisco ASR 1006 Router, can have two RP ports, wherein one RP is active and the other is standby. Both the active and standby RP BITS ports can be configured as output sources.

Step 6	**exit** **Example:** `Router(config)# exit`	Exits the global configuration mode.

Examples

The following example shows how to configure clock recovery from a BITS port for a Metronome SPA:

```
Router# configure terminal
Router(config)# network-clock synchronization automatic
Router(config)# network-clock synchronization ssm option 2 GEN1
Router(config)# network-clock input-source 1 external 0/3/0 t1 esf
Router(config)# exit
```

The following example shows how to configure clock recovery from a BITS port for RP BITS on Cisco ASR 1000 Series Aggregation Services Routers with dual RP:

```
Router# configure terminal
Router(config)# network-clock synchronization automatic
Router(config)# network-clock synchronization ssm option 2 GEN
Router(config)# network-clock input-source 1 External R0 t1 esf
Router(config)# exit
```

The following example shows how to configure clock recovery with BITS as the input source on the Cisco ASR 1002-X Series Aggregation Services Routers on the E1 Common Channel Signal mode:

```
Router# configure terminal
Router(config)# network-clock synchronization automatic
Router(config)# network-clock synchronization ssm option 2 GEN
Router(config)# network-clock input-source 1 external r0 e1 cas crc4 120 linecode
Router(config)# exit
```

Configuring Clock Recovery with a BITS Port as an Output-Source

This section describes how to configure clock recovery from a BITS port as an output-source.

Note BITS configuration as output sources is supported only on Cisco ASR 1002-X Router.

SUMMARY STEPS

1. **configure terminal**
2. **network-clock synchronization automatic**
3. **network-clock synchronization ssm option** *option-id generation-id*
4. **network-clock output-source** *priority* **{external** *slot/card/port***}**
5. **exit**

DETAILED STEPS

	Command	Purpose
Step 1	**enable** **Example:** `Router# enable`	Enables the privileged EXEC mode. Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters the global configuration mode.
Step 3	**network-clock synchronization automatic** **Example:** `Router(config)# network-clock synchronization automatic`	Enables the network clock selection algorithm. This command disables the Cisco-specific network clock process, and turns on the G.781-based automatic clock selection process.
Step 4	**network-clock synchronization ssm option {1 | 2 {GEN1 | GEN2}}** **Example:** `Router(config)# network-clock synchronization ssm option 2 GEN1`	Configures the equipment to work in synchronization network. The *option_id* value 1 refers to synchronization networks design for Europe. This is the default value. The *option_id* value 2 refers to the synchronization networks design for the U.S.

	Command	Purpose
Step 5	**network-clock output-source priority {external** slot/card/port**}** **Example:** Router(config)# **network-clock output-source system 1 External 0/3/0 t1 esf** Router(config)# **network-clock output-source 1 External R0 t1 esf** **Example:For ASR 1002-X Router:** Router(config)# **network-clock output-source 1 external r0 e1 cas crc4 120 linecode**	Enables clock recovery from a BITS port. **Note** For routers with dual hardware support such as the Cisco ASR 1006 Router can have two RP ports wherein one RP is active and the other is standby. Both the active and standby RP BITS ports can be configured as output sources.
Step 6	**exit** **Example:** Router(config)# **exit**	Exits the global configuration mode.

Examples

The following example shows how to configure clock recovery from a BITS port for a Metronome SPA:

```
Router# configure terminal
Router(config)# network-clock synchronization automatic
Router(config)# network-clock synchronization ssm option 2 GEN1
Router(config)# network-clock output-source 1 external 0/3/0 t1 esf
Router(config)# exit
```

The following example shows how to configure clock recovery from a BITS port for RP BITS on Cisco ASR 1000 Series Aggregation Services Routers with dual RP:

```
Router# configure terminal
Router(config)# network-clock synchronization automatic
Router(config)# network-clock synchronization ssm option 2 GEN
Router(config)# network-clock output-source 1 external R0 t1 esf
Router(config)# exit
```

The following example shows how to configure clock recovery with BITS as output-source on Cisco ASR 1002-X Series Aggregation Services Routers on a E1 Common Channel Signal Mode:

```
Router# configure terminal
Router(config)# network-clock synchronization automatic
Router(config)# network-clock synchronization ssm option 2 GEN
Router(config)# network-clock output-source 1 external r0 e1 cas crc4 120 linecode
Router(config)# exit
```

Configuring SyncE by Using the Line to External Feature

This section describes how to configure network synchronization by using the Line to External feature.

Note The Line to External feature is not supported on the Cisco ASR 1002-X Router.

SUMMARY STEPS

1. **configure terminal**
2. **network-clock synchronization automatic**
3. **network-clock synchronization ssm option {1 | 2 {GEN1 | GEN2}}**
4. **interface gigabitethernet** *slot/card/port*
5. **synchronous mode**
6. **exit**
7. **network-clock output-source line** *priority* **{interface** *interface-name***} {external** *slot/card/port***}**
8. **exit**

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal** **Example:** `Router# configure terminal`	Enters the global configuration mode.
Step 2	**network-clock synchronization automatic** **Example:** `Router(config)# network-clock synchronization automatic`	Enables the network clock selection algorithm. This command disables the Cisco-specific network clock process, and turns on the G.781-based automatic clock selection process.
Step 3	**network-clock synchronization ssm option {**`1 | 2` **{GEN1 | GEN2}}** **Example:** `Router(config)# network-clock synchronization ssm option 2 GEN1`	Configures the equipment to work in a synchronization network. The *option_id* value 1 refers to a synchronization network design for Europe. This is the default value. The *option_id* value 2 refers to a synchronization network design for the U.S.
Step 4	**interface gigabitethernet** *slot/card/port* **Example:** `Router(config)# interface gigabitethernet 0/2/0`	Specifies the Gigabit Ethernet interface to be configured: *slot/card/port*—Specifies the location of the interface.

	Command	Purpose
Step 5	**synchronous mode** **Example:** `Router(config-if)# synchronous mode`	Changes to the synchronous mode.
Step 6	**exit** **Example:** `Router(config)# exit`	Exits the specific configuration mode.
Step 7	**network-clock output-source line** *priority* **{interface** *interface-name***} {external** *slot/card/port***}** **Example:** `Router(config-if-srv)# network-clock output-source line 1 interface gigabitethernet 0/2/0 external 0/3/0`	Configures the line clock to be used on external Tx interfaces. **Note** This command is supported only on the Metronome SPA.
Step 8	**exit** **Example:** `Router(config)# exit`	Exits the global configuration mode.

Examples

The following example shows how to configure a SyncE using the Line to External method:

```
Router# configure terminal
Router(config)# network-clock synchronization automatic
Router(config)# network-clock synchronization ssm option 2 GEN1
Router(config)# interface gigabitethernet 0/2/0
Router(config-if)# synchronous mode
Router(config)# exit
Router(config)# network-clock output-source line 1 interface gigabitethernet 0/2/0
external 0/3/0
Router(config)# exit
```

Managing Synchronization

Manage synchronization on a Cisco ASR 1000 Series Aggregation Service Router using the following management commands:

- **network-clock synchronization mode QL-enabled** command

  Use the **network-clock synchronization mode QL-enabled** command in the global configuration mode to configure the automatic selection process for the QL-enabled mode. This succeeds only if the SyncE interfaces are capable of sending SSMs.

  The following example shows how to configure network clock synchronization (QL-enabled mode) in the global configuration mode:

  ```
  Router(config)# network-clock synchronization mode QL-enabled
  ```

- **esmc process** command

  Use the **esmc process** command in the global configuration mode to enable the ESMC process at system level. The **no** form of this command disables the ESMC process. The **no** form of this command fails if no SyncE-capable interface is installed on the platform.

  The following example shows how to enable the ESMC process in the global configuration mode:

  ```
  Router(config)# esmc process
  ```

- **esmc mode [tx | rx]** command

  Use the **esmc mode [tx | rx]** command in the interface configuration mode to enable the ESMC process at the interface level. The **no** form of the command disables the ESMC process.

  The following example shows how to enable ESMC in the interface configuration mode:

  ```
  Router(config-if)# esmc mode tx
  ```

- **network-clock quality-level** command

  Use the **network-clock source quality-level** command in the interface configuration mode to configure the QL value for ESMC on a *gigabit Ethernet* port. The value is based on global interworking options:

  - If Option 1 is configured, the available values are QL-PRC, QL-SSU-A, QL-SSU-B, QL-SEC, and QL-DNU.
  - If Option 2 is configured with GEN 2, the available values are QL-PRS, QL-STU, QL-ST2, QL-TNC, QL-ST3, QL-SMC, QL-ST4, and QL-DUS.
  - If Option 2 is configured with GEN1, the available values are QL-PRS, QL-STU, QL-ST2, QL-SMC, QL-ST4, and QL-DUS

  The following example shows how to configure the **network-clock source quality-level** in the interface configuration mode:

  ```
  Router(config-if)# network-clock source quality-level QL-PRC rx
  ```

  Use the **network-clock quality-level** command in the global configuration mode to configure the QL value for the SSMs on a BITS port.

  The following example shows how to configure **network-clock quality-level** command in the global configuration mode:

  ```
  Router(config)# network-clock quality-level rx qL-pRC external R0 2048k
  ```

- **network-clock wait-to-restore** command

  Use the **network-clock wait-to-restore** *timer* **global** command to set the wait-to-restore time. You can configure the wait-to-restore time to any value between 0 to 86400 seconds. The default value is 300 seconds. The wait-to-restore timer can be set in the global configuration mode and the interface configuration mode.

  The following example shows how to configure the wait-to-restore timer in the global configuration mode:

```
Router(config)# network-clock wait-to-restore 10 global
```

  The following example shows how to configure the wait-to-restore timer in the interface configuration mode:

```
Router(config)# interface gigabitethernet 0/2/0
Router(config-if)# network-clock wait-to-restore 10
```

- **network-clock hold-off** command

  Use the **network-clock hold-off** *timer* **global** command to configure hold-off time. You can configure the hold-off time to either 0 or any value between 50 to 10000 ms. The default value is 300 ms. The **network-clock hold-off** *timer* can be set in the global configuration mode and the interface configuration mode.

  The following example shows how to configure the hold-off timer:

```
Router(config-if)# network-clock hold-off 50 global
```

- **network-clock switch force** command

  Use the **network-clock switch force** command to forcefully select a synchronization source irrespective of whether the source is available, and within range.

  The following example shows how to configure a force switch:

```
Router# network-clock switch force interface gigabitethernet 0/2/0
```

- **network-clock switch manual** command

  Use the **network-clock switch manual** command to manually select a synchronization source, provided the clock source has a higher quality level than the currently active clock.

  The following example shows how to configure a manual switch:

```
Router# network-clock switch manual interface gigabitethernet 0/2/0
```

- **network-clock clear switch controller-id** command

  Use the **network-clock clear switch controller-id** command to clear the manual, or switch it on by force.

  The following example shows how to clear a switch:

```
Router# network-clock clear switch t0
```

- **network-clock set lockout** command

  Use the **network-clock set lockout** command to lockout a clock source. A clock source flagged as lockout is not selected for SyncE. To clear the lock-out on a source, use the **network-clock clear lockout** command.

  The following example shows how to lock out a clock source:

```
Router# network-clock set lockout interface gigabitethernet 0/2/0
```

The following example shows how to clear the lock out on a clock source:

```
Router# network-clock clear lockout interface gigabitethernet 0/2/0
```

Verifying the Network Synchronization Configuration

Use the following commands to verify the network synchronization configuration:

- Use the **show network-clock synchronization** command to display the output:

```
Router# show network-clock synchronization

Symbols:     En - Enable, Dis - Disable, Adis - Admin Disable
             NA - Not Applicable
             *  - Synchronization source selected
             #  - Synchronization source force selected
             &  - Synchronization source manually switched

Automatic selection process : Enable
Equipment Clock : 1544 (EEC-Option2)
Clock Mode : QL-Enable
ESMC : Enabled
SSM Option : GEN1
T0 : External R0 t1 esf
Hold-off (global) : 300 ms
Wait-to-restore (global) : 300 sec
Tsm Delay : 180 ms
Revertive : No
Force Switch: FALSE
Manual Switch: FALSE
Number of synchronization sources: 2
sm(netsync NETCLK_QL_ENABLE), running yes, state 1A
Last transition recorded: (src_rem)-> 1A (src_added)-> 1A (src_rem)-> 1A (src_added)->
1A (ql_change)-> 1A (sf_change)-> 1A (force_sw)-> 1C (clear_sw)-> 1A (sf_change)-> 1A
(sf_change)-> 1A


Nominated Interfaces

 Interface            SigType     Mode/QL      Prio  QL_IN  ESMC Tx  ESMC Rx
Internal             NA          NA/Dis       251   QL-ST3    NA        NA
Gi1/2/0              NA          Sync/En      1     QL-FAILED -         -
*External R0          T1 ESF      NA/En        2     QL-STU    NA        NA
```

- Use the **show network-clock synchronization detail** command to display all the details of network clock synchronization parameters at the global and interface level, as shown in the following example:

```
Router# show network-clock synchronization detail

Symbols:     En - Enable, Dis - Disable, Adis - Admin Disable
             NA - Not Applicable
             *  - Synchronization source selected
             #  - Synchronization source force selected
             &  - Synchronization source manually switched

Automatic selection process : Enable
Equipment Clock : 1544 (EEC-Option2)
Clock Mode : QL-Enable
ESMC : Enabled
SSM Option : GEN1
```

```
T0 : External R0 t1 esf
Hold-off (global) : 300 ms
Wait-to-restore (global) : 300 sec
Tsm Delay : 180 ms
Revertive : No
Force Switch: FALSE
Manual Switch: FALSE
Number of synchronization sources: 2
sm(netsync NETCLK_QL_ENABLE), running yes, state 1A
Last transition recorded: (src_rem)-> 1A (src_added)-> 1A (src_rem)-> 1A (src_added)->
1A (ql_change)-> 1A (sf_change)-> 1A (force_sw)-> 1C (clear_sw)-> 1A (sf_change)-> 1A
(sf_change)-> 1A


Nominated Interfaces

Interface            SigType     Mode/QL      Prio  QL_IN  ESMC Tx  ESMC Rx
Internal             NA          NA/Dis       251   QL-ST3    NA        NA
Gi1/2/0              NA          Sync/En      1     QL-FAILED -         -
*External R0          T1 ESF      NA/En        2     QL-STU    NA        NA

Interface:
-----------------------------------------------
Local Interface: Internal
Signal Type: NA
Mode: NA(Ql-enabled)
SSM Tx: Disable
SSM Rx: Disable
Priority: 251
QL Receive: QL-ST3
QL Receive Configured: -
QL Receive Overrided: -
QL Transmit: -
QL Transmit Configured: -
Hold-off: 0
Wait-to-restore: 0
Lock Out: FALSE
Signal Fail: FALSE
Alarms: FALSE
Slot Disabled: FALSE

Local Interface: Gi1/2/0
Signal Type: NA
Mode: Synchronous(Ql-enabled)
ESMC Tx: Disable
ESMC Rx: Enable
Priority: 1
QL Receive: QL-PRS
QL Receive Configured: -
QL Receive Overrided: QL-FAILED
QL Transmit: QL-DUS
QL Transmit Configured: -
Hold-off: 300
Wait-to-restore: 300
Lock Out: FALSE
Signal Fail: FALSE
Alarms: TRUE
Slot Disabled: FALSE

Local Interface: External R0
Signal Type: T1 ESF
Mode: NA(Ql-enabled)
SSM Tx: Disable
SSM Rx: Enable
```

```
Priority: 2
QL Receive: QL-STU
QL Receive Configured: -
QL Receive Overrided: -
QL Transmit: -
QL Transmit Configured: -
Hold-off: 300
Wait-to-restore: 300
Lock Out: FALSE
Signal Fail: FALSE
Alarms: FALSE
Slot Disabled: FALSE
```

- Use the **show esmc** command to display the ESMC configuration output:

```
Router# show esmc
Interface: GigabitEthernet0/0/0
  Administative configurations:
    Mode: Synchronous
    ESMC TX: Enable
    ESMC RX: Enable
    QL TX: -
    QL RX: -
    Operational status:
    Port status: UP
    QL Receive: QL-PRC
    QL Transmit: QL-DNU
    QL rx overrided: -
    ESMC Information rate: 1 packet/second
    ESMC Expiry: 5 second
Interface: GigabitEthernet0/0/0
  Administrative configurations:
    Mode: Synchronous
    ESMC TX: Enable
    ESMC RX: Enable
    QL TX: -
    QL RX: -
    Operational status:
    Port status: UP
    QL Receive: QL-DNU
    QL Transmit: QL-DNU
    QL rx overrided: QL-DNU
    ESMC Information rate: 1 packet/second
    ESMC Expiry: 5 second
```

- Use the **show esmc detail** command to display all the details of the ESMC parameters at the global and interface level, as shown in the following example:

```
Router# show esmc detail
Router#show esmc detail
Interface: GigabitEthernet0/0/0
  Administative configurations:
    Mode: Synchronous
    ESMC TX: Enable
    ESMC RX: Enable
    QL TX: -
    QL RX: QL-PRS
  Operational status:
    Port status: UP
    QL Receive: QL-DUS
    QL Transmit: QL-ST3
    QL rx overrided: QL-DUS
```

```
ESMC Information rate: 1 packet/second
ESMC Expiry: 5 second
ESMC Tx Timer: Running
ESMC Rx Timer: Running
ESMC Tx interval count: 1
ESMC INFO pkts in: 0
ESMC INFO pkts out: 256
ESMC EVENT pkts in: 0
ESMC EVENT pkts out: 0
```

Troubleshooting the Network Synchronization Configuration

Note Before you troubleshoot, ensure that all the network synchronization configurations are complete.

Table 11-4 provides information about troubleshooting scenarios that you may encounter while configuring network synchronization.

Table 11-4 Troubleshooting Scenarios

Problem	Solution
Incorrect clock selection	• Verify that there are no alarms on the interfaces. Use the **show network-clock synchronization detail** command to check this. • Ensure that the nonrevertive configurations are in place.
Incorrect QL values	Ensure that there is no framing mismatch with the SSM option.

Table 11-4 Troubleshooting Scenarios (continued)

Problem	Solution

Incorrect clock limit set or queue limit disabled mode	• Verify that there are no alarms on the interfaces. Use the **show network-clock synchronization detail** RP command to confirm.
	• Use the **show network-clock synchronization** command to confirm if the system is in revertive mode or nonrevertive mode and verify the nonrevertive configurations as shown in this example:
	`Router#show network-clock synchronization`
	`Symbols: En - Enable, Dis - Disable, Adis - Admin Disable`
	`NA - Not Applicable`
	`* - Synchronization source selected`
	`# - Synchronization source force selected`
	`& - Synchronization source manually switched`
	
	`Automatic selection process : Enable`
	`Equipment Clock : 1544 (EEC-Option2)`
	`Clock Mode : QL-Enable`
	`ESMC : Enabled`
	`SSM Option : GEN1`
	`T0 : External R0 t1 esf`
	`Hold-off (global) : 300 ms`
	`Wait-to-restore (global) : 300 sec`
	`Tsm Delay : 180 ms`
	`Revertive : No`
	`Force Switch: FALSE`
	`Manual Switch: FALSE`
	`Number of synchronization sources: 2`
	`sm(netsync NETCLK_QL_ENABLE), running yes, state 1A`
	`Last transition recorded: (src_rem)-> 1A (src_added)-> 1A (src_rem)-> 1A (src_added)-> 1A (ql_change)-> 1A (sf_change)-> 1A (force_sw)-> 1C (clear_sw)-> 1A (sf_change)-> 1A (sf_change)-> 1A`

Incorrect QL values observed when you use the **show network-clock synchronization detail** command	Use the **network-clock synchronization SSM** (*option 1* |*option 2*) command to confirm that there is no framing mismatch. Use the **show run interface** command to validate the framing for a specific interface. For SSM *option 1*, framing should be SDH or E1, and for SSM *option 2*, it should be T1.
Mismatched Physical line coding mechanisms between clock source and input source on the router	To ensure that this feature works correctly, both the ends need to be configured with identical physical line coding mechanism and encapsulation values.

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	*Cisco IOS Master Commands List, All Releases*

Standards

Standard	Title
None	—

MIBs

MIB	MIBs Link
None	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at this URL: http://www.cisco.com/go/mibs

RFCs

RFC	Title
None	—

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for Network Synchronization Support

Table 11-5 lists the features in this module and provides links to specific configuration information.

Use Cisco Feature Navigator to find information about platform support and software image support. Cisco Feature Navigator enables you to determine which software images support a specific software release, feature set, or platform. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Note Table 11-5 lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Table 11-5 Feature Information for Network Synchronization Support

Feature Name	Releases	Feature Information
Network Synchronization Support	Cisco IOS XE 3.2S	In Cisco IOS XE Release 3.2S, this feature was introduced on the Cisco ASR 1000 Series Aggregation Service Routers.
Network Synchronization Support	Cisco IOS XE 3.7S	In Cisco IOS XE Release 3.7S, the commands that are used to configure this feature have been extended with the introduction of the Cisco ASR 1002-X Router.
Network Synchronization Support	Cisco IOS XE 3.12S	In Cisco IOS XE Release 3.12S, the commands that are used to configure this feature have been extended to the Cisco ASR 1001-X Router.

IEEE 1588v2 PTP Support

IEEE 1588v2 Precision Time Protocol (PTP) is a packet-based two-way message exchange protocol for synchronizing clocks between nodes in a network, thereby enabling an accurate time distribution over a network.This document explains how to configure IEEE 1588v2 PTP on the Cisco ASR 1002-X Routers.

Finding Feature Information

Your software release might not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the "Feature Information for IEEE 1588v2 PTP Support" section on page 12-21.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Contents

This guide covers the following topics:

- Restrictions for IEEE 1588v2 PTP, page 12-1
- Information About IEEE 1588v2 PTP, page 12-2
- Configuring IEEE 1588v2 PTP, page 12-7
- Additional References, page 12-19
- Feature Information for IEEE 1588v2 PTP Support, page 12-21

Restrictions for IEEE 1588v2 PTP

These are the restrictions for configuring IEEE 1588v2 PTP:

- Supports IPv4 unicast mode, but not multicast mode.
- Does not support Dot1q, Q-in-Q, and port-channel interfaces.
- PTP master supports only a maximum of 32 PTP slaves.
- PTP boundary clock is supported only in unicast negotiation mode.

- IPv6 and Multiprotocol Label Switching (MPLS) encapsulation are not supported for PTP packet transfer over Cisco ASR 1002-X Routers.
- The time-of-day recovered from a 1588v2 session does not synchronize with the system clock.
- GPS interfaces can be used only for clock recovery. You cannot transmit the system clock on the GPS interface.

Information About IEEE 1588v2 PTP

IEEE 1588v2 PTP is a packet-based two-way message exchange protocol for synchronizing a local clock with a primary reference clock or a grand master clock in hierarchical master-slave architecture. This synchronization is achieved through packets that are transmitted and received in a session between a master clock and a slave clock. IEEE 1588v2 PTP supports system-wide synchronization accuracy in the sub-microsecond range with little use of network and local clock-computing resources.

The following sections describe the terminologies used for better understanding of the IEEE 1588v2 PTP.

PTP Clocks

PTP employs a hierarchy of clock types to ensure that precise timing and synchronization is maintained between the source and the numerous PTP clients that are distributed throughout the network. A logical grouping of PTP clocks that synchronize with each other using the PTP protocol, but are not necessarily synchronized to the PTP clocks in another domain, is called a PTP domain.

The three PTP clock types are Ordinary clock, Boundary clock, and Transparent clock.

- **Ordinary clock--**This clock type has a single PTP port in a domain, and maintains the timescale used in the domain. It may serve as a source of time, that is, be a master, or may synchronize to another clock by being a slave. It provides time to an application or to an end device.
- **Boundary clock--**This clock type has multiple PTP ports in a domain, and maintains the timescale used in the domain. It may serve as a source of time, that is, be a master, or may synchronize to another clock by being a slave. A boundary clock that is a slave has a single slave port, and transfers timing from that port to the master ports.
- **Transparent clock--**This clock type is a device that measures the time taken for a PTP event message to pass through the device, and provides this information to the clocks receiving this PTP event message.

Table 15-1 shows the 1588v2 PTP support matrix on a Cisco ASR1000 platform.

Table 12-1 1588v2 PTP Support Matrix on a Cisco ASR1000 platform

Platform/PTP Clock mode	Ordinary Clock	Boundary Clock	Transparent Clock	Hybrid Clock
ASR1002X	Yes	Yes	No	No

Components of a PTP-enabled Network

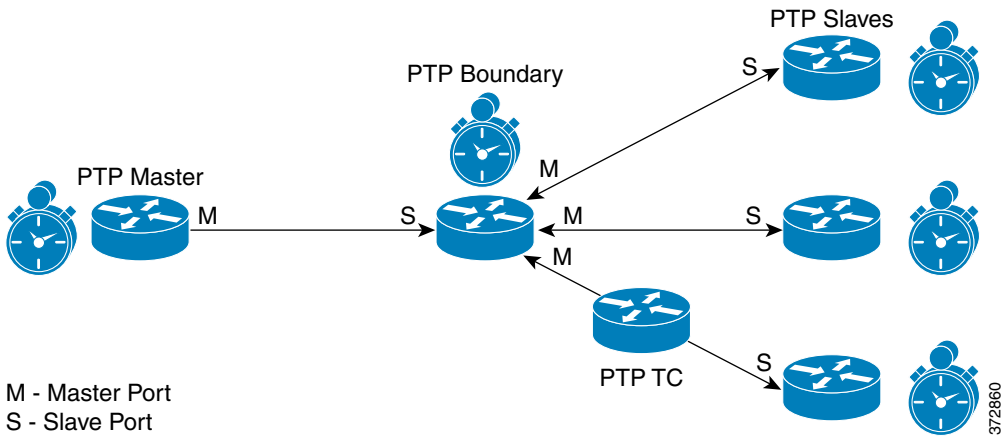
The three key components of a PTP-enabled data network are grand master, PTP client, and PTP-enabled router acting as a Boundary clock.

- **Grand Master**--An IEEE1588v2 PTP network needs a grand master to provide a precise time source. The most economical way of obtaining the precise time source for the grand master is through a Global Positioning System (GPS) because it provides +/- 100 nanosecond (ns) accuracy. First, the PTP grand master's built-in GPS receiver converts the GPS timing information to PTP time information, which is typically Coordinated Universal Time (UTC), and then delivers the UTC time to all the PTP clients.
- **PTP client**--A PTP client has to be installed on servers, network-monitoring and performance-analysis devices, or other devices that want to use the precise timing information provided by PTP, and it's mostly an ordinary clock. The two kinds of PTP clients are pure software PTP clients and hardware-assistant PTP clients.
- **PTP boundary clock**--Any router that is between a PTP master and PTP slave can act as a PTP boundary clock router. It has two interfaces, one facing the PTP master and another facing the PTP slave. The boundary clock router acts as a *slave* on the interface facing the *PTP master router*, and acts as a *master* on the interface facing the *PTP slave router*. The PTP boundary clock router is deployed to minimize timing delay in cases where the distance between PTP master router and the PTP slave router is more.

Note Intermediary nodes between PTP master and slave should be a PTP-enabled or transparent clock node.

Figure 12-1 shows the functions of a PTP Enabled device.

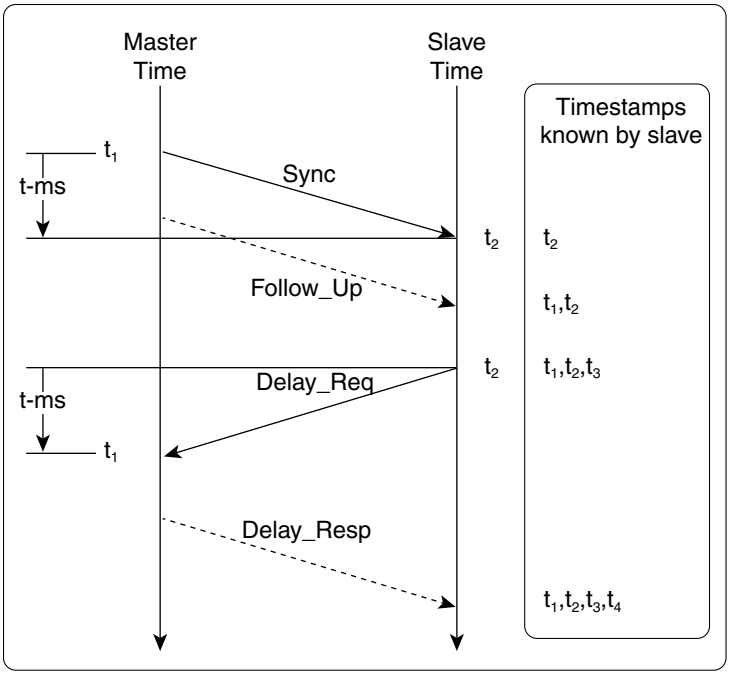
Figure 12-1 Functions of a PTP-Enabled Device



Clock-Synchronization Process

Clock synchronization is achieved through a series of messages exchanged between the master clock and the slave clock as shown in Figure 12-2.

Figure 12-2 Clock-Synchronization Process



After the master-slave clock hierarchy is established, the clock synchronization process starts. The message exchange occurs in this sequence:

1. The master clock sends a Sync message. The time at which the Sync message leaves the master is time-stamped as t_1.
2. The slave clock receives the Sync message and is time-stamped as $t_{2.}$.
3. The slave sends the Delay_Req message, which is time-stamped as t_3 when it leaves the slave, and as t_4 when the master receives it.
4. The master responds with a Delay_Resp message that contains the time stamp t_4.

The clock offset is the difference between the master clock and the slave clock, and is calculated as follows:

$$\text{Offset} = t_2 - t_1 - \text{meanPathDelay}$$

IEEE1588 assumes that the path delay between the master clock and the slave clock is symmetrical, and hence, the mean path delay is calculated as follows:

$$\text{meanPathDelay} = ((t_2 - t_1) + (t_4 - t_3))/2$$

PTP Messages

All PTP communication is performed through message exchange. The two sets of messages defined by IEEE1588v2 are General messages and Event messages.

- **General messages**--These messages do not require accurate time stamps, and are classified as Announce, Follow_Up, Delay_Resp, Pdelay_Resp_Follow_Up, Management, and Signaling.
- **Event messages**--These messages require accurate time stamping, and are classified as Sync, Delay_Req, Pdelay_Req, and Pdelay_Resp.

PTP Clocking Modes

The following are the PTP clocking modes supported on a Cisco ASR 1002-X Router:

- **Unicast Mode**--In unicast mode, the master sends the Sync or Delay_Resp messages to the slave on the unicast IP address of the slave, and the slave in turn sends the Delay_Req message to the master on the unicast IP address of the master.
- **Unicast Negotiation Mode**--In unicast negotiation mode, the master does not know of any slave until the slave sends a negotiation message to the master. The unicast negotiation mode is good for scalability purpose because one master can have multiple slaves.

PTP Accuracy

Accuracy is an important aspect of PTP implementation on an Ethernet port. For a packet network, Packet Delay Variation (PDV) is one of the key factors that impacts the accuracy of a PTP clock. The Cisco ASR 1002-X Router can handle the PDV of the network with its advanced hardware and software capabilities, such as hardware stamping and special high-priority queue for PTP packets. It can provide around 300 ns accuracy in a scalable deployment scenario.

The two methods used on the same topology to cross-check and verify the results are:

- One-pulse-per-second (1PPS) to verify the PTP slave.
- Maximum Time Interval Error (MTIE) and Time Deviation (TDEV) to verify the PDV.

The verification topology includes a grand master with a GPS receiver, a Cisco ASR 1002-X Router, PTP hardware slave clocks with 1PPS output, and a test equipment for the measurement.

Figure 12-3 1PPS Accuracy Measurement

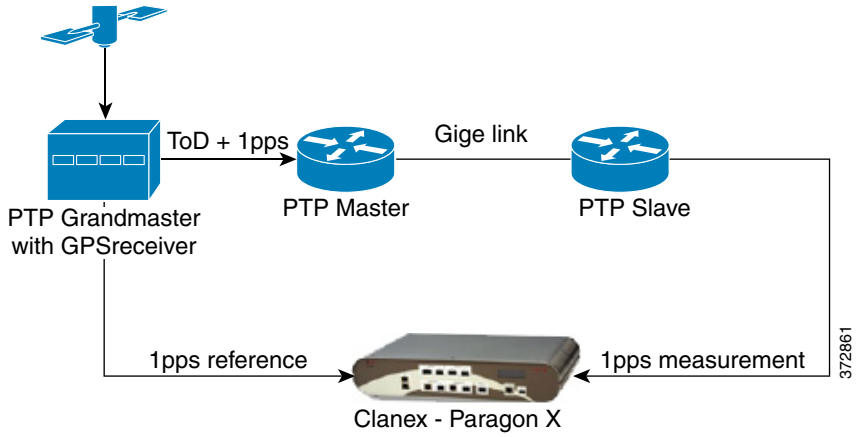


Figure 12-4 shows the PPS accuracy, with time of day measured using the test equipment as per the topology shown in Figure 12-3. The average PPS accuracy value found is 250 ns.

Figure 12-4 Graph Showing PPS Accuracy

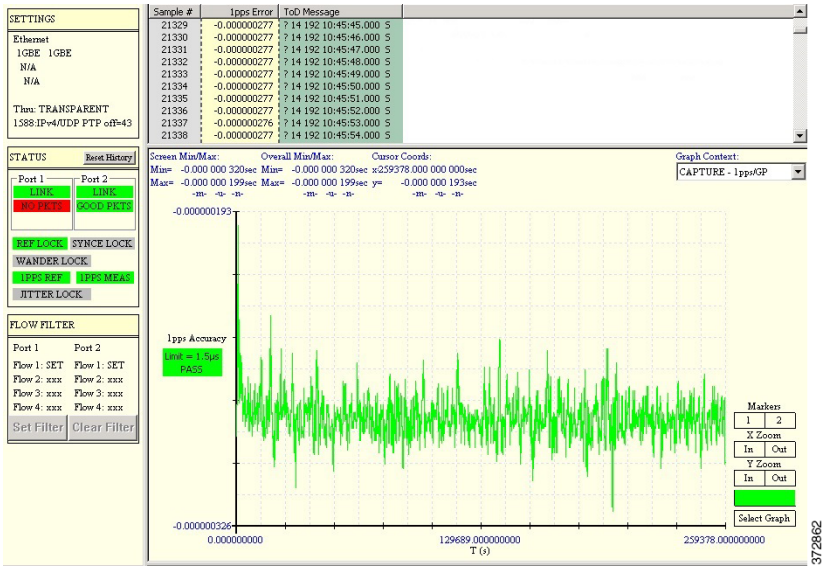


Figure 12-5 shows a topology that includes a grand master with a GPS receiver, a Cisco ASR 1002-X Router, PTP hardware slave clocks, and a test equipment for the MTIE and TDEV measurement.

Figure 12-5 MTIE and TDEV measurement

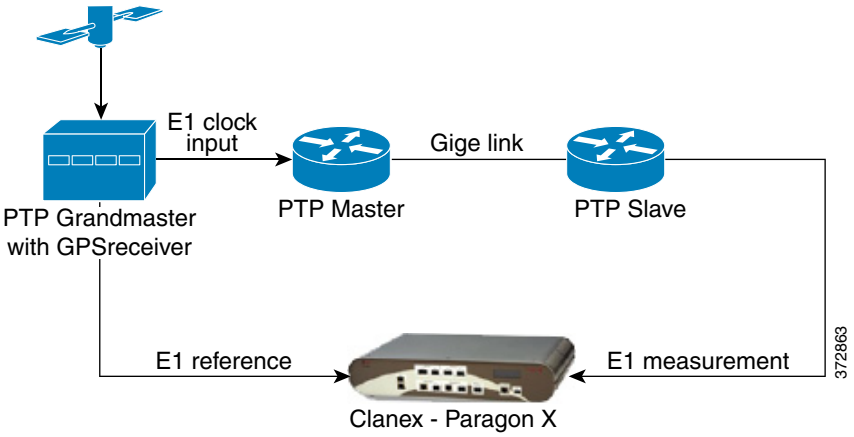
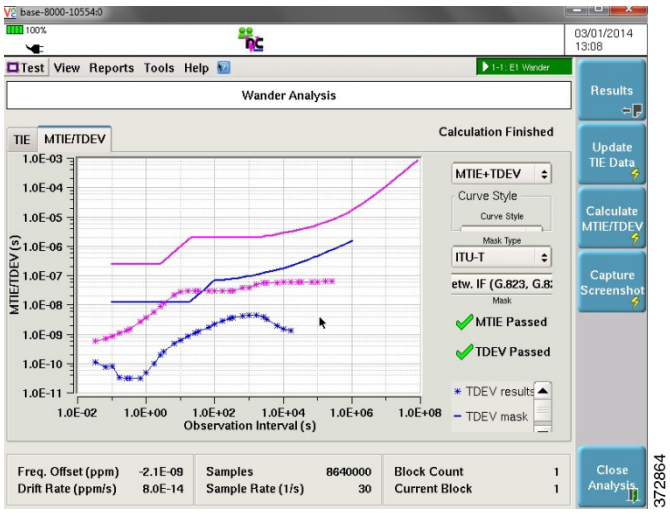


Figure 12-6 shows a graph with the MTIE and TDEV measurements to verify the PDV.

Figure 12-6 Graph to show MTIE and TDEV Measurement



IEEE 1588v2 PTP Support

IEEE 1588v2 PTP supports these features on a Cisco ASR1002-X Router:

- Two-step Ordinary clock and Boundary clock.
- Hardware-assistant PTP implementation to provide sub-300 ns accuracy.
- PTP operation on all physical onboard Gigabit Ethernet interfaces.
- Supports built-in Gigabit Ethernet links in two-step clock mode.

Configuring IEEE 1588v2 PTP

You can configure IEEE 1588v2 PTP features on the Cisco ASR 1002-X Router by performing the following procedures:

- Configuring Input or Output Network Clocking, page 12-8
- Configuring an Ordinary Clock, page 12-8
- Configuring a Boundary Clock, page 12-11
- Configuring Time of Day, page 12-13
- Configuration Examples for IEEE 1588v2 PTP on a Cisco ASR1002-X Router, page 12-17
- Verifying the IEEE 1588v2 PTP Configuration, page 12-18

Configuring Input or Output Network Clocking

We recommend that you configure a stable input clock source from a GPS device before configuring PTP master. The GPS device acts as a PTP grand master, and the BITS or 10-MHz port of a Cisco ASR 1002-X Router can be used to input or output the network clock. Perform these tasks to configure network clocking on a Cisco ASR 1002-X Router:

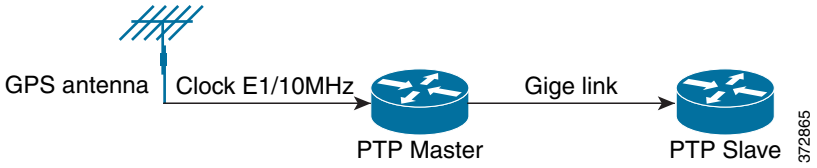
- Configuring Input Clocking
- Configuring Output Clocking

See the module Synchronous Ethernet Support for more information on network clocking.

Configuring an Ordinary Clock

You can configure a Cisco ASR 1002-X Router in Ordinary clock mode as either master or slave.

Figure 12-7 Ordinary Clock Scenario with a GPS Device as Grand Master



Perform these tasks to configure an ordinary clock as either master or slave:

- Configuring an Ordinary Clock as PTP Master, page 12-8
- Configuring an Ordinary Clock as PTP Slave, page 12-10

Configuring an Ordinary Clock as PTP Master

This section describes how to configure an ordinary clock as PTP master.

SUMMARY STEPS

1. **configure terminal**
2. **ptp clock ordinary domain** *domain_number*
3. **clock-port** *name* **master**
4. **transport ipv4 unicast interface {GigabitEthernet | Loopback}** *interface-number* **{negotiation}**
5. **clock destination** *ip-address*
6. **sync interval** *interval*
7. **end**

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 2	**ptp clock ordinary domain** *domain_number* **Example:** `Router(config)# ptp clock ordinary domain 0`	Creates a PTP clock and specifies the clock mode.
Step 3	**clock-port** *name* **master** **Example:** `Router(config-ptp-clk)# clock-port MASTER master`	Specifies the clocking mode of a PTP port and enters the clock port configuration mode.
Step 4	**transport ipv4 unicast interface {GigabitEthernet | Loopback}** *interface-number* **[negotiation]** **Example:** `Router(config-ptp-port)# transport ipv4 unicast interface Loopback11 negotiation`	Specifies the IP version, transmission mode, and interface that a PTP clock port uses to exchange timing packets. The **negotiation** keyword specifies the unicast negotiation mode where the slave and master clock exchange negotiation messages before establishing a relationship. **Note** Only Loopback interface type is supported.
Step 5	**clock destination** *ip-address* **Example:** `Router(config-ptp-port)# clock destination 20.20.20.20`	Specifies the IP address of a PTP clock destination. If the clock port is set to master mode with unicast negotiation, you need not use this command because the device uses negotiation to determine the IP address of PTP slave devices.
Step 6	**sync interval** *interval* **Example:** `Router(config-ptp-port)# sync interval -4`	(Optional) Specifies the interval used to send PTP synchronization messages. The default value is -5.
Step 7	**end** **Example:** `Router(config-ptp-port)# end`	Exits global configuration mode.

Examples

The following example shows how to configure an ordinary clock as PTP master:

```
Router# configure terminal
Router(config)# ptp clock ordinary domain 0
Router(config-ptp-clk)# clock-port MASTER master
```

```
Router(config-ptp-port)# transport ipv4 unicast interface Loopback11 negotiation
Router(config-ptp-port)# clock destination 20.20.20.20
Router(config-ptp-port)# Sync interval -4
Router(config-ptp-port)# end
```

Configuring an Ordinary Clock as PTP Slave

This section describes how to configure Ordinary Clock as PTP slave.

SUMMARY STEPS

1. **configure terminal**
2. **ptp clock ordinary domain** *domain_number*
3. **clock-port** *name* **slave**
4. **transport ipv4 unicast interface {GigabitEthernet | Loopback}** *interface-number* **{negotiation}**
5. **clock source** *ip-address*
6. **sync interval** *interval*
7. **end**

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 2	**ptp clock ordinary domain** *domain_number* **Example:** `Router(config)# ptp clock ordinary domain 0`	Creates a PTP clock and specifies the clock mode.
Step 3	**clock-port** *name* **slave** **Example:** `Router(config-ptp-clk)# clock-port SLAVE slave`	Specifies the clocking mode of a PTP port and enters the clock port configuration mode.
Step 4	**transport ipv4 unicast interface {GigabitEthernet | Loopback}** *interface-number* **[negotiation]** **Example:** `Router(config-ptp-port)# transport ipv4 unicast interface Loopback22 negotiation`	Specifies the IP version, transmission mode, and interface that a PTP clock port uses to exchange timing packets. The **negotiation** keyword specifies the unicast negotiation mode where the slave and master clock exchanges negotiation messages before establishing a relationship. **Note** Only Loopback interface type is supported.

	Command	Purpose
Step 5	**clock source** *ip-address* **Example:** `Router(config-ptp-port)# clock source 10.10.10.10`	Specifies the source IP address of a PTP master clock. **Note** You can specify only 1 master clock IP address. Priority-based clock source selection is not supported.
Step 6	**end** **Example:** `Router(config-ptp-port)# end`	Exits global configuration mode.

Examples

The following example shows how to configure an ordinary clock as PTP slave:

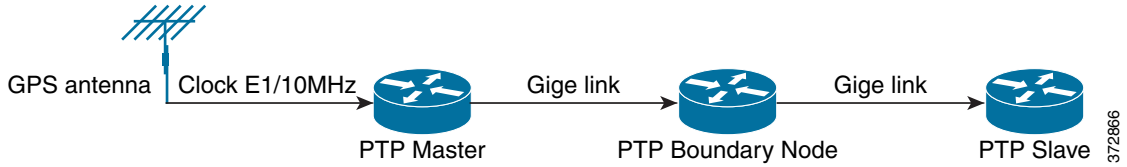
```
Router# configure terminal
Router(config)# ptp clock ordinary domain 0
Router(config-ptp-clk)# clock-port SLAVE master
Router(config-ptp-port)# transport ipv4 unicast interface Loopback22 negotiation
Router(config-ptp-port)# clock source 10.10.10.10
Router(config-ptp-port)# end
```

Configuring a Boundary Clock

You can configure the PTP master and PTP slave in a boundary clock topology as shown in Figure 12-8 in the same way that you configure a master and slave in ordinary clock mode. This section describes how to configure a Cisco ASR 1002-X Router in boundary clock mode.

Note Currently, boundary clock supports only unicast negotiation mode.

Figure 12-8 PTP Boundary Clock Scenario



SUMMARY STEPS

1. **configure terminal**
2. **ptp clock boundary domain** *domain_number*
3. **clock-port** *name* **slave**
4. **transport ipv4 unicast interface {GigabitEthernet | Loopback}** *interface-number* **{negotiation}**
5. **clock source** *ip-address*

6. **exit**
7. **clock-port** *name* **master**
8. **transport ipv4 unicast interface {GigabitEthernet | Loopback}** *interface-number* **{negotiation}**
9. **end**

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal** **Example:** `Router# configure terminal`	Enters the global configuration mode.
Step 2	**ptp clock boundary domain** *domain_number* **Example:** `Router(config)# ptp clock boundary domain 0`	Creates a PTP clock and specifies the clock mode.
Step 3	**clock-port** *name* **slave** **Example:** `Router(config-ptp-clk)# clock-port SLAVE slave`	Specifies the clocking mode of a PTP port and enters the clock port configuration mode.
Step 4	**transport ipv4 unicast interface {GigabitEthernet | Loopback}** *interface-number* **[negotiation]** **Example:** `Router(config-ptp-port)# transport ipv4 unicast interface Loopback11 negotiation`	Specifies the IP version, transmission mode, and interface that a PTP clock port uses to exchange timing packets. The **negotiation** keyword specifies the unicast negotiation mode where the slave and master clock exchange negotiation messages before establishing a relationship. **Note** Only Loopback interface type is supported.
Step 5	**clock source** *ip-address* **Example:** `Router(config-ptp-port)# clock source 10.10.10.10`	Specifies the source IP address of a PTP master clock. **Note** You can specify only one master clock IP address. Priority-based clock source selection is not supported.
Step 6	**exit** **Example:** `Router(config-ptp-port)# exit`	Exits clock port configuration mode.
Step 7	**clock-port** *name* **master** **Example:** `Router(config-ptp-clk)# clock-port MASTER master`	Specifies the clocking mode of a PTP port and enters clock port configuration mode.

	Command	Purpose
Step 8	**transport ipv4 unicast interface {GigabitEthernet | Loopback}** *interface-number* **[negotiation]** **Example:** `Router(config-ptp-port)# transport ipv4 unicast interface Loopback10 negotiation`	Specifies the IP version, transmission mode, and interface that a PTP clock port uses to exchange timing packets. The **negotiation** keyword specifies the unicast negotiation mode where the slave and master clock exchange negotiation messages before establishing a relationship. **Note** Only Loopback interface type is supported.
Step 9	**end** **Example:** `Router(config-ptp-port)# end`	Exits global configuration mode.

Examples

The following example shows how to configure a boundary clock:

```
Router# configure terminal
Router(config)# ptp clock ordinary domain 0
Router(config-ptp-clk)# clock-port SLAVE slave
Router(config-ptp-port)# transport ipv4 unicast interface Loopback11 negotiation
Router(config-ptp-port)# clock source 10.10.10.10
Router(config-ptp-port)# exit
Router(config-ptp-clk)# clock-port MASTER master
Router(config-ptp-port)# transport ipv4 unicast interface Loopback10 negotiation
Router(config-ptp-port)# end
```

Configuring Time of Day

A Cisco ASR 1002-X Router can exchange time of day and 1PPS input with an external device, such as a GPS receiver, using the time of day and 1PPS input and output interfaces on the router.

Perform these tasks to configure Time of Day (ToD) messages on the Cisco ASR 1002-X Router:

- Configuring Input Time-of-Day Messages, page 12-13
- Configuring Output Time-of-Day Messages, page 12-15

Configuring Input Time-of-Day Messages

This section describes how to configure input time-of-day messages.

Note You can configure time-of-day input only in a PTP master clock port.

SUMMARY STEPS

1. **configure terminal**
2. **ptp clock ordinary domain** *domain_number*
3. **tod{R0 | R1} {cisco | ntp}**

4. **input [1pps] {R0|R1}**
5. **clock-port** *name* **master**
6. **transport ipv4 unicast interface {GigabitEthernet | Loopback}** ***interface-number*** **{negotiation}**
7. **clock destination** *ip-address*
8. **end**

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 2	**ptp clock ordinary domain** *domain_number* **Example:** `Router(config)# ptp clock ordinary domain 0`	Creates a PTP clock and specifies the clock mode.
Step 3	**tod {R0 | R1} {cisco | ntp}** **Example:** `Router(config-ptp-clk)# tod R0 ntp`	Configures the time-of-day message format used by the 1PPS or BITS interface. **Note** Currently, only R0 1PPS port is supported; R1 is not valid. Also, only ntp mode is supported, not cisco mode.
Step 4	**input [1pps] { R0 | R1 }** **Example:** `Router(config-ptp-clk)# input 1pps R0`	Enables PTP input clocking using a 1.544-Mhz, 2.048-Mhz, or 10-Mhz timing interface, or phase using the 1PPS or RS-422 interface. **Note** Currently, only R0 1PPS port is supported; R1 is not valid.
Step 5	**clock-port** *name* **master** **Example:** `Router(config-ptp-clk)# clock-port MASTER master`	Specifies the clocking mode of a PTP port and enters the clock port configuration mode.
Step 6	**transport ipv4 unicast interface {GigabitEthernet | Loopback}** *interface-number* **[negotiation]** **Example:** `Router(config-ptp-port)# transport ipv4 unicast interface Loopback11 negotiation`	Specifies the IP version, transmission mode, and interface that a PTP clock port uses to exchange timing packets. The **negotiation** keyword specifies the unicast negotiation mode where the slave and master clock exchange negotiation messages before establishing a relationship. **Note** Only Loopback interface type is supported.

	Command	Purpose
Step 7	**clock destination** *ip-address* **Example:** `Router(config-ptp-port)# clock destination 20.20.20.20`	Specifies the IP address of a PTP clock destination. If the clock port is set to master mode with unicast negotiation, you need not use this command because the device uses negotiation to determine the IP address of PTP slave devices.
Step 8	**end** **Example:** `Router(config-ptp-port)# end`	Exits global configuration mode.

Examples

The following example shows how to configure input time-of-day messages:

```
Router# configure terminal
Router(config)# ptp clock ordinary domain 0
Router(config-ptp-clk)# tod R0 ntp
Router(config-ptp-clk)# input 1pps R0
Router(config-ptp-clk)# clock-port MASTER master
Router(config-ptp-port)# transport ipv4 unicast interface Loopback11 negotiation
Router(config-ptp-port)# clock destination 20.20.20.20
Router(config-ptp-port)# end
```

Configuring Output Time-of-Day Messages

This section describes how to configure output time-of-day messages.

Note You can configure ToD output only on PTP slave clock ports.

SUMMARY STEPS

1. **configure terminal**
2. **ptp clock ordinary domain** *domain_number*
3. **tod{R0 | R1} {cisco | ntp}**
4. **output [1pps] {R0|R1}**
5. **clock-port** *name* **slave**
6. **transport ipv4 unicast interface {GigabitEthernet | Loopback}** ***interface-number*** **{negotiation}**
7. **clock source** *ip-address*
8. **end**

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 2	**ptp clock ordinary domain** *domain_number* **Example:** `Router(config)# ptp clock ordinary domain 0`	Creates a PTP clock and specifies the clock mode.
Step 3	**tod {R0 | R1} {cisco | ntp}** **Example:** `Router(config-ptp-clk)# tod R0 ntp`	Configures the time-of-day message format used by the 1PPS or BITS interface. **Note** Currently, only R0 1PPS port is supported; R1 is not valid. Also, only ntp mode is supported, not cisco mode.
Step 4	**output [1pps] { R0 | R1 }** **Example:** `Router(config-ptp-clk)# output R0 ntp`	Enables output of time-of-day messages using a 1PPS interface. **Note** Currently, only R0 1PPS port is supported; R1 is not valid.
Step 5	**clock-port** *name* **slave** **Example:** `Router(config-ptp-clk)# clock-port SLAVE slave`	Specifies the clocking mode of a PTP port and enters the clock port configuration mode.
Step 6	**transport ipv4 unicast interface {GigabitEthernet | Loopback}** *interface-number* **[negotiation]** **Example:** `Router(config-ptp-port)# transport ipv4 unicast interface Loopback11 negotiation`	Specifies the IP version, transmission mode, and interface that a PTP clock port uses to exchange timing packets. The **negotiation** keyword specifies the unicast negotiation mode where the slave and master clock exchange negotiation messages before establishing a relationship. **Note** Only Loopback interface type is supported.
Step 7	**clock source** *ip-address* **Example:** `Router(config-ptp-port)# clock source 10.10.10.10`	Specifies the source IP address of a PTP master clock. **Note** You can specify only 1 master clock IP address. Priority-based clock source selection is not supported.
Step 8	**end** **Example:** `Router(config-ptp-port)# end`	Exits global configuration mode.

Examples

The following example shows how to configure output time-of-day messages:

```
Router# configure terminal
Router(config)# ptp clock ordinary domain 0
Router(config-ptp-clk)# tod R0 ntp
Router(config-ptp-clk)# output 1pps R0
Router(config-ptp-clk)# clock-port MASTER master
Router(config-ptp-port)# transport ipv4 unicast interface Loopback11 negotiation
Router(config-ptp-port)# clock source 10.10.10.10
Router(config-ptp-port)# end
```

Configuration Examples for IEEE 1588v2 PTP on a Cisco ASR1002-X Router

This example shows how to configure IEEE 1588v2 PTP on a Cisco ASR1002-X Router:

Unicast Negotiation Mode

Master Clock

```
ptp clock ordinary domain 1
tod R0 ntp
input 1pps R0
clock-port  MASTER master
transport ipv4 unicast interface loopback 0 negotiation
```

Slave clock

```
ptp clock ordinary domain 1
tod R0 ntp
output 1pps R0
clock-port SLAVE slave
transport ipv4 unicast interface loopback 0 negotiation
clock source 10.1.1.1
```

Boundary clock

```
ptp clock boundary domain 1
clock-port SLAVE slave
transport ipv4 unicast interface loopback 0 negotiation
clock source 10.1.1.1
clock-port  MASTER master
transport ipv4 unicast interface loopback 1 negotiation
```

Unicast Mode

Master Clock

```
ptp clock ordinary domain 1
tod R0 ntp
input 1pps R0
clock-port  MASTER master
transport ipv4 unicast interface loopback 0
clock destination 20.1.1.1
```

Slave clock

```
ptp clock ordinary domain 1
tod R0 ntp
output 1pps R0
clock-port SLAVE slave
```

```
transport ipv4 unicast interface loopback 0
clock source 10.1.1.1
```

Verifying the IEEE 1588v2 PTP Configuration

Use the following commands to verify the IEEE 1588v2 PTP configuration:

- Use the **show ptp clock running domain 0** command to display the output:

```
Router# show ptp clock running domain 0

On the MASTER:
                     PTP Ordinary Clock [Domain 0]

         State           Ports          Pkts sent      Pkts rcvd      Redundancy Mode

         FREQ_LOCKED     1              31522149       10401171       Hot standby

                               PORT SUMMARY
                                                                        PTP Master
Name    Tx Mode      Role         Transport    State        Sessions     Port Addr

MASTER unicast       master       Lo1          Master       1            -


                              SESSION INFORMATION

MASTER [Lo1] [Sessions 1]

Peer addr           Pkts in    Pkts out   In Errs    Out Errs

11.11.11.11         10401171   31522149   0          0

On the SLAVE:

                     PTP Ordinary Clock [Domain 0]

         State           Ports          Pkts sent      Pkts rcvd      Redundancy Mode

         PHASE_ALIGNED   1              4532802        13357682       Track one

                               PORT SUMMARY
                                                                          PTP Master
Name   Tx Mode      Role         Transport    State        Sessions     Port Addr

SLAVE unicast       slave        Lo20         Slave        1            10.10.10.10


                              SESSION INFORMATION

SLAVE [Lo20] [Sessions 1]

Peer addr           Pkts in    Pkts out   In Errs    Out Errs

10.10.10.10         13357682   4532802    0          0
```

- Use the **show platform software ptp tod** command to check the time-of-day information:

```
PTPd ToD information:

Time: 06/24/14 02:06:29
```

- Use the **show platform ptp tod all** command to check the time-of- day state:

```
Router# show platform ptp tod all

On the MASTER
--------------------------------
ToD/1PPS Info for : R0

--------------------------------
RJ45 JACK TYPE        : RS422
ToD CONFIGURED        : YES
ToD FORMAT            : NTPv4
ToD DELAY             : 0
1PPS MODE             : INPUT
1PPS STATE            : UP
ToD STATE             : UP
--------------------------------
On the SLAVE:
--------------------------------
ToD/1PPS Info for : R0
--------------------------------
RJ45 JACK TYPE        : RS422
ToD CONFIGURED        : YES
ToD FORMAT            : NTPv4
ToD DELAY             : 0
1PPS MODE             : OUTPUT
OFFSET                : 0
PULSE WIDTH           : 0
--------------------------------
```

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	*Cisco IOS Master Commands List, All Releases*

MIBs

MIB	MIBs Link
None	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at this URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for IEEE 1588v2 PTP Support

Table 12-2 lists the features in this module and provides links to specific configuration information.

Use Cisco Feature Navigator to find information about platform support and software image support. Cisco Feature Navigator enables you to determine which software images support a specific software release, feature set, or platform. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Note Table 12-2 lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Table 12-2 Feature Information for Network Synchronization Support

Feature Name	Releases	Feature Information
IEEE 1588v2 PTP Support	Cisco IOS XE 3.13S	In Cisco IOS XE Release 3.13S, this feature was introduced on the Cisco ASR 1002-X Routers.

Configuring Bridge Domain Interfaces

The Cisco ASR 1000 Series Aggregation Services Routers support the bridge domain interface (BDI) feature for packaging Layer 2 Ethernet segments into Layer 3 IP.

Finding Feature Information

Your software release might not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the "Feature Information for Configuring Bridge Domain Interfaces" section on page 13-11.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Contents

- Restrictions for Bridge Domain Interfaces, page 13-1
- Information About Bridge Domain Interfaces, page 13-2
- Creating or Deleting a Bridge Domain Interface, page 13-6
- Bridge Domain Interface Scalability, page 13-6
- How to Configure a Bridge Domain Interface, page 13-6
- Additional References, page 13-10
- Feature Information for Configuring Bridge Domain Interfaces, page 13-11

Restrictions for Bridge Domain Interfaces

The following are the restrictions pertaining to bridge domain interfaces:

- Only 4096 bridge domain interfaces are supported per system.
- Bridge domain interfaces do not support the following features:
  - PPP over Ethernet (PPPoE)
  - Bidirectional Forwarding Detection (BFD) protocol

  - Netflow
  - QOS
  - Network-Based Application Recognition (NBAR) or Advanced Video Coding (AVC)
- Bridge domain interfaces support the following features from IOS XE 3.8.2 and later:
  - Hot Standby Router Protocol (HSRP)
  - Virtual Router Redundancy Protocol (VRRP)
- For a BDI, the maximum transmission unit (MTU) size can be configured between 1500 and 9216 bytes.
- Cryptographic VPNs are not supported in combination with BDI.
- MPLS is not supported on bridge domain interfaces.

Information About Bridge Domain Interfaces

Bridge domain interface is a logical interface that allows bidirectional flow of traffic between a Layer 2 bridged network and a Layer 3 routed network traffic. Bridge domain interfaces are identified by the same index as the bridge domain. Each bridge domain represents a Layer 2 broadcast domain. Only one bridge domain interface can be associated with a bridge domain.

Bridge domain interface supports the following features:

- IP termination
- Layer 3 VPN termination
- Address Resolution Protocol (ARP), G-ARP, and P-ARP handling
- MAC address assignment

Prior to configuring a bridge domain interface, you must understand the following concepts:

- Ethernet Virtual Circuit Overview, page 13-2
- Bridge Domain Interface Encapsulation, page 13-3
- Assigning a MAC Address, page 13-3
- Support for IP Protocols, page 13-3
- Support for IP Forwarding, page 13-4
- Packet Forwarding, page 13-4
- Bridge Domain Interface Statistics, page 13-5

Ethernet Virtual Circuit Overview

An Ethernet Virtual Circuit (EVC) is an end-to-end representation of a single instance of a Layer 2 service being offered by a provider to a customer. It embodies the different parameters on which the service is being offered. In the Cisco EVC Framework, the bridge domains are made up of one or more Layer 2 interfaces known as service instances. A service instance is the instantiation of an EVC on a given port on a given router. Service instance is associated with a bridge domain based on the configuration.

An incoming frame can be classified as service instance based on the following criteria:

- Single 802.1Q VLAN tag, priority-tagged, or 802.1ad VLAN tag
- Both QinQ (inner and outer) VLAN tags, or both 802.1ad S-VLAN and C-VLAN tags
- Outer 802.1p CoS bits, inner 802.1p CoS bits, or both
- Payload Ethernet type (five choices are supported: IPv4, IPv6, PPPoE-all, PPoE-discovery, and PPPoE-session)

Service instance also supports alternative mapping criteria:

- Untagged—Mapping to all the frames lacking a 802.1Q or 802.1ad header
- Default—Mapping to all the frames

For more information on the EVC architecture, see Configuring Ethernet Virtual Connections on the Cisco ASR 1000 Router chapter in the *Carrier Ethernet Configuration Guide* located at:

http://www.cisco.com/en/US/docs/ios-xml/ios/cether/configuration/xe-3s/ce-ether-vc-infra-xe.html

Bridge Domain Interface Encapsulation

An EVC provides the ability to employ different encapsulations on each Ethernet flow point (EFP) present in a bridge domain. A BDI egress point may not be aware of the encapsulation of an egress packet because the packet may have egressed from one or more EFPs with different encapsulations.

In a bridge domain, if all the EFPs have different encapsulations, the BDI must be untagged (using the no 802.1Q tag). Encapsulate all the traffic in the bridge domain (popped or pushed) at the EFPs. Configure **rewrite** at each EFP to enable encapsulation of the traffic on the bridge domain.

In a bridge domain, if all the EFPs have the same encapsulation, configure the encapsulations on the BDI using the **encapsulation** command. Enabling encapsulation at the BDI ensures effective pushing or popping of tags, thereby eliminating the need for configuring the **rewrite** command at the EFPs. For more information on configuring the encapsulations on the BDI, see the How to Configure a Bridge Domain Interface, page 13-6.

Assigning a MAC Address

All the bridge domain interfaces on the Cisco ASR 1000 chassis share a common MAC address. The first bridge domain interface on a bridge domain is allocated a MAC address. Thereafter, the same MAC address is assigned to all the bridge domain interfaces that are created in that bridge domain.

Note You can configure a static MAC address on a bridge domain interface using the **mac-address** command.

Support for IP Protocols

Brigde domain interfaces enable the Cisco ASR 1000 Series Aggregation Services Routers to act as a Layer 3 endpoint on the Layer 2 bridge domain for the following IP-related protocols:

- ARP
- DHCP
- HTTP
- ICMP

- NTP
- RARP
- SNMP
- TCP
- Telnet
- TFTP
- UDP

Support for IP Forwarding

Bridge domain interface supports the following IP forwarding features:

- IPv4 input and output access control lists (ACL)
- IPv4 input and output QoS policies. The operations supported for the input and output service policies on a bridge domain interface are:
  - Classification
  - Marking
  - Policing
- IPv4 L3 VRFs

Packet Forwarding

A bridge domain interface provides bridging and forwarding services between the Layer 2 and Layer 3 network infrastructure.

Layer 2 to Layer 3

During a packet flow from a Layer 2 network to a Layer 3 network, if the destination MAC address of the incoming packet matches the bridge domain interface MAC address, or if the destination MAC address is a multicast address, the packet or a copy of the packet is forwarded to the bridge domain interface.

Note MAC address learning cannot not be performed on the bridge domain interface.

Note In a bridge domain, when flooding unknown unicast frames, bridge domain interface is not included.

Layer 3 to Layer 2

When a packet arrives at a Layer 3 physical interface of a router, a route lookup action is performed. If route lookup points to a bridge domain interface, then the bridge domain interface adds the layer 2 encapsulation and forwards the frame to the corresponding bridge domain. The byte counters are updated.

During a Layer 2 lookup on a bridge domain to which the bridge domain interface belongs, the bridge domain forwards the packets to the correct service instance based on the destination MAC address.

Link States of a Bridge Domain and a Bridge Domain Interface

Bridge domain interface acts as a routable IOS interface on Layer 3 and as a port on a bridge domain. Both bridge domain interfaces and bridge domains operate with individual administrative states.

Shutting down a bridge domain interface stops the Layer 3 data service, but does not override or impact the state of the associated bridge domain.

Shutting down a bridge domain stops Layer 2 forwarding across all the associated members including service instances and bridge domain interfaces. The operational state of a bridge domain is influenced by associated service instances. Bridge domain interface cannot be operational unless one of the associated service instance is up.

Note Because a bridge domain interface is an internal interface, the operational state of bridge domain interface does not affect the bridge domain operational state.

BDI Initial State

The initial administrative state of a BDI depends on how the BDI is created. When a BDI is created at boot time in the startup configuration, the default administrative state for the BDI will be up, and will remain in this state unless the startup configuration includes the **shutdown** command. This behavior is consistent with all the other interfaces. When a BDI is created dynamically by a user at command prompt, the default administrative state is down.

BDI Link State

As with all Cisco IOS interfaces, a BDI maintains a link state that comprises of three states, administratively down, operationally down, and up. The link state of a BDI is derived from two independent inputs, the BDI administrative state set by the corresponding users and the fault indication state from the lower levels of the interface states. defines a BDI link state based on the state of the two inputs.

Fault Indication State	BDI Administrative State	
—	Shutdown	No Shutdown
No faults asserted	Admin-down	Up
At least one fault asserted	Admin-down	Operationally-down

Bridge Domain Interface Statistics

For virtual interfaces, such as the bridge domain interface, protocol counters are periodically queried from the QFP.

When packets flow from a Layer 2 bridge domain network to a Layer 3 routing network through the bridge domain interface, the packets are treated as bridge domain interface input packets and bytes. When packets arrive at a Layer 3 interface and are forwarded through the bridge domain interface to a Layer 2 bridge domain, the packets are treated as output packets and bytes, and the counters are updated accordingly.

A BDI maintains a standard set of Layer 3 packet counters as the case with all Cisco IOS interfaces. Use the **show interface** command to view the Layer 3 packet counters.

The convention of the counters is relative to the Layer 3 cloud, for example, *input* refers to the traffic entering the Layer 3 cloud from the Layer 2 BD, while *output* refers to the traffic leaving the Layer 3 cloud to the Layer 2 BD.

Use the **show interfaces accounting** command to display the statistics for the BDI status. Use the **show interface** <*if-name*> command to display the overall count of the packets and bytes that are transmitted and received.

Creating or Deleting a Bridge Domain Interface

When you define an interface or subinterface for a Cisco IOS router, you name it and specify how it is assigned an IP address.You can create a bridge domain interface before adding a bridge domain to the system, this new bridge domain interface will be activated after the associated bridge domain is configured.

Note When a bridge domain interface is created, a bridge domain is automatically created.

When both bridge domain interface and bridge domain are created, the system maintains the required associations for mapping the bridge domain-bridge domain interface pair.

The mapping of bridge domain and bridge domain interface is maintained in the system. The bridge domain interface uses the index of the associated bridge domain to show the association.

Bridge Domain Interface Scalability

Table 13-1 lists the bridge domain interface scalability numbers, based on the type of Cisco ASR 1000 Series Aggregation Services Router's Forwarding Processors.

Table 13-1 Bridge Domain Interface Scalability Numbers Based on the Type of Cisco ASR 1000 Series Aggregation Services Router's Forwarding Processors

Description	ASR1000-ESP5, ASR 1001, ASR 1002-F (ESP2.5)	ASR1000-ESP10, ASR1000-ESP10-N, ASR1000-ESP20	ASR1000-ESP40
Maximum bridge domain interfaces per router	4096	4096	4096

How to Configure a Bridge Domain Interface

To configure a bridge domain interface, perform the following steps:

SUMMARY STEPS

1. **configure terminal**
2. **interface BDI** *interface number*
3. **encapsulation** *encapsulation dot1q <first-tag> [second-dot1q <second-tag>]*
4. **ip address** *ip-address mask*
5. **mac-address** *{mac-address}*
6. **no shut**
7. **shut**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Router# configure terminal`	Enters the global configuration mode.
Step 2	**interface BDI** {interface number} **Example:** `Router(config)# interface BDI3`	Specifies a bridge domain interface on a Cisco ASR 1000 Series Aggregation Services Router.
Step 3	**encapsulation** encapsulation dot1q <first-tag> [second-dot1q <second-tag>] **Example:** `Router(config-if)#encapsulation dot1Q 1 second-dot1q 2`	Defines the encapsulation type. The example shows how to define dot1q as the encapsulation type.
Step 4	**ip address** *ip-address mask* or **ipv6 address** {*X:X:X:X::X* **link-local** | *X:X:X:X::X/prefix* [**anycast** | **eui-64**] | **autoconfig** [**default**]} **Example:** `Router(config-if)# ip address 2.2.2.1 255.255.255.0`	Specifies either the IPv4 or IPv6 address for the bridge domain interface.
Step 5	**mac-address** *{mac-address}* `Router(config-if)# mac-address 1.1.3`	Specifies the MAC address for the bridge domain interface.
Step 6	**no shut** **Example:** `hostname(config-if)# no shut`	Enables the bridge domain interface on the Cisco ASR 1000 Series Aggregation Services Router.
Step 7	**shut** **Example:** `hostname(config-if)# shut`	Disables the bridge domain interface on the Cisco ASR 1000 Series Aggregation Services Router.

Example

The following example shows the configuration of a bridge domain interface at IP address 2.2.2.1 255.255.255.0:

```
Router# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)# interface BDI3
Router(config-if)# encapsulation dot1Q 1 second-dot1q 2
Router(config-if)# ip address 2.2.2.1 255.255.255.0
Router(config-if)# mac-address 1.1.3
Router(config-if)# no shut
```

```
Router(config-if)# exit
Router(config)#
```

You can use different **show** commands to display the bridge domain interface configuration information:

- **show interfaces bdi—**Displays the configuration summary of the corresponding BDI.

  Example:

  ```
  Router# show interfaces BDI3
  ```

- **show platform software interface fp active name**—Displays the bridge domain interface configuration in a Forwarding Processor.

  Example:

  ```
  Router# show platform software interface fp active name BDI4
  ```

- **show platform hardware qfp active interface if-name—**Displays the bridge domain interface configuration in a data path.

  Example:

  ```
  Router# show platform hardware qfp active interface if-name BDI4
  ```

You can use the following debug commands to debug the bridge domain interface configurations:

- **debug platform hardware qfp feature**—Debugs the features in the Cisco Quantum Flow Processor (QFP).

  The following example shows how to debug the l2bd for all the clients:

  ```
  Router# debug platform hardware qfp active feature l2bd client all
  The selected CPP L2BD Client debugging is on.
  ```

- **platform trace runtime process forwarding-manager module**—Enables the Forwarding Manager Route Processor and Embedded Service Processor trace messages for the Forwarding Manager process.

  In the following example, the trace level for the forwarding processor module in the Forwarding Manager of the ESP processor in slot 0 is set to the informational tracing level (info):

  ```
  Router(config)# platform trace runtime slot F0 bay 0 process forwarding-manager module
  interfaces level info
  ```

- **platform trace boottime process forwarding-manager module interfaces**—Enables the Forwarding Manager Route Processor and Embedded Service Processor trace messages for the Route Processor Forwarding Manager process during bootup.

  In the following example, the trace level for the forwarding processor module in the Forwarding Manager of the ESP processor in slot R0 is set to the informational tracing level (max):

  ```
  Router(config)# platform trace boottime slot R0 bay 1 process forwarding-manager
  forwarding-manager level max
  ```

For additional information on the commands and the options available with each command, see the *Cisco IOS Configuration Fundamentals Command Reference Guide* located at:

http://www.cisco.com/en/US/docs/ios/fundamentals/command/reference/cf_book.html

Additional References

Related Documents

Related Topic	Document Title
Configuring Ethernet Virtual Connections on the Cisco ASR 1000 Series Aggregation Services Routers	*Carrier Ethernet Configuration Guide, Cisco IOS XE Release 3S*
EVC Quality of Service	http://www.cisco.com/en/US/docs/ios/ios_xe/qos/configuration/guide/qos_evc_xe.html
Cisco IOS commands	*Cisco IOS Master Commands List, All Releases*

Standards

Standard	Title
None	—

MIBs

MIB	MIBs Link
None	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at this URL: http://www.cisco.com/go/mibs

RFCs

RFC	Title
None	—

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for Configuring Bridge Domain Interfaces

Table 13-2 lists the features in this module and provides links to specific configuration information.

Use Cisco Feature Navigator to find information about platform support and software image support. Cisco Feature Navigator enables you to determine which software images support a specific software release, feature set, or platform. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Note Table 13-2 lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Table 13-2 Feature Information for Configuring Bridge Domain Interfaces

Feature Name	Releases	Feature Information
Configuring Bridge Domain Interface	Cisco IOS XE 3.2.0S	This feature was introduced on the Cisco ASR 1000 Series Aggregation Services Routers.
Configuring Bridge Domain Interface	Cisco IOS XE 3.7.0S	This feature was updated on the Cisco ASR 1000 Series Aggregation Services Routers. The following section was updated for this feature: Information About Bridge Domain Interfaces, page 13-2

Enabling Support for Tunable DWDM-XFP-C

First Published: July 2013

The dense wavelength-division multiplexing (DWDM) wavelengths of the DWDM-XFP-C module on the Cisco ASR 1000 Series Aggregation Services Routers is tunable. You can configure the DWDM ITU wavelengths using the **itu channel** command in the interface configuration mode. The **itu channel** command ensures that the traffic continues to flow.

Table 14-1 contains the wavelength mapping information for the DWDM-XFP-C. module

Table 14-1 DWDM-XFP-C Wavelength Mapping

Channel	Frequency (THz)	Wavelength (nm)
1	191.95	1561.83
2	192.00	1561.42
3	192.05	1561.01
4	192.10	1560.61
5	192.15	1560.20
6	192.20	1559.79
7	192.25	1559.39
8	192.30	1558.98
9	192.35	1558.58
10	192.40	1558.17
11	192.45	1557.77
12	192.50	1557.36
13	192.55	1556.96
14	192.60	1556.55
15	192.65	1556.15
16	192.70	1555.75
17	192.75	1555.34
18	192.80	1554.94
19	192.85	1554.54
20	192.90	1554.13

Table 14-1 DWDM-XFP-C Wavelength Mapping (continued)

Channel	Frequency (THz)	Wavelength (nm)
21	192.95	1553.73
22	193.00	1553.33
23	193.05	1552.93
24	193.10	1552.52
25	193.15	1552.12
26	193.20	1551.72
27	193.25	1551.32
28	193.30	1550.92
29	193.35	1550.52
30	193.40	1550.12
31	193.45	1549.72
32	193.50	1549.32
33	193.55	1548.91
34	193.60	1548.51
35	193.65	1548.11
36	193.70	1547.72
37	193.75	1547.32
38	193.80	1546.92
39	193.85	1546.52
40	193.90	1546.12
41	193.95	1545.72
42	194.00	1545.32
43	194.05	1544.92
44	194.10	1544.53
45	194.15	1544.13
46	194.20	1543.73
47	194.25	1543.33
48	194.30	1542.94
49	194.35	1542.54
50	194.40	1542.14
51	194.45	1541.75
52	194.50	1541.35
53	194.55	1540.95
54	194.60	1540.56
55	194.65	1540.16
56	194.70	1539.77

Table 14-1 DWDM-XFP-C Wavelength Mapping (continued)

Channel	Frequency (THz)	Wavelength (nm)
57	194.75	1539.37
58	194.80	1538.98
59	194.85	1538.58
60	194.90	1538.19
61	194.95	1537.79
62	195.00	1537.40
63	195.05	1537.00
64	195.10	1536.61
65	195.15	1536.22
66	195.20	1535.82
67	195.25	1535.43
68	195.30	1535.04
69	195.35	1534.64
70	195.40	1534.25
71	195.45	1533.86
72	195.50	1533.47
73	195.55	1533.07
74	195.60	1532.68
75	195.65	1532.29
76	195.70	1531.90
77	195.75	1531.51
78	195.80	1531.12
79	195.85	1530.72
80	195.90	1530.33
81	195.95	1529.94
82	196.00	1529.55

Configuring the DWDM-XFP-C Module

Perform the following procedure to configure the DWDM-XFP-C module.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface tengigabitethernet** *slot/port*
4. **itu channel** *number*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** *source-interface gigabitethernet slot/port* **Example:** `Router# enable`	Enables the privileged EXEC mode. If prompted, enter your password.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters the global configuration mode.
Step 3	**interface tengigabitethernet** *slot/port* *Router(config)# interface tengigabitethernet 4/11*	Specifies the 10-Gigabit Ethernet interface to be configured. Here: *slot/port*—Specifies the location of the interface.
Step 4	**itu channel** *number* **Example:** *Router(config-if)# itu channel 28*	Sets the ITU channel. *number*—Specifies the ITU channel number. The acceptable values are from 1–82.

Verifying the ITU Configuration

The following example shows how to use the **show hw-module subslot** command to check an ITU configuration:

```
Router# enable
Router# conf t
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#int tenGigabitEthernet 1/2/0
Router(config-if)#itu channel 20
Router(config-if)#
Router#
Router# show hw-module subslot 1/2 transceiver 0 idpr dump
Description = XFP optics (type 6)
Transceiver Type: = TUNABLE DWDM XFP (194)
Product Identifier (PID) = DWDM-XFP-C
Frequency Set for Tunable DWDM = 195.5 THz
```

```
Vendor Revision = 00
Serial Number (SN) = JFX1617800W
Vendor Name = CISCO-JDSU
Vendor OUI (IEEE company ID) = 00.01.9C (412)
CLEI code = IP9IAGGCAB
Cisco part number = 10-2544-02
Device State = Disabled.
XFP IDPROM Page 0x0:
000: 0C 00 49 00 F8 00 46 00 FB 00
010: 00 00 00 00 00 00 00 00 A6 04
020: 09 C4 8C A0 13 88 9B 83 13 93
030: 62 1F 1F 07 0F 8D 00 0A 09 CF
040: 00 10 00 18 FF E8 00 0C FF F4
050: 00 00 00 00 00 00 00 00 00 00
060: 00 BF 25 1C 00 C4 00 00 01 F4
070: 00 00 00 00 00 00 00 00 00 00
080: 00 00 00 00 9E 20 00 00 00 00
090: 00 00 00 00 00 00 1E 7C 00 00
100: 00 00 00 01 00 00 00 00 00 00
110: E2 98 00 14 00 00 00 00 00 00 <<See byte 113, the hexa decimal
equivalent for ITU channel 20>>
120: 00 00 00 00 00 00 00 01
XFP IDPROM Page 0x1:
128: 0C 98 07 00 00 00 00 00 00 00
138: 08 B4 63 71 50 00 00 00 00 9F
148: 43 49 53 43 4F 2D 4A 44 53
```

Monitoring and Maintaining Multilink Frame Relay

The Cisco Frame Relay implementation currently supports routing on IP, DECnet, AppleTalk, XNS, Novell IPX, CLNS, Banyan VINES, and transparent bridging. This guide describes how to monitor and maintain the Multilink Frame Relay feature for the Cisco ASR 1000 Series Routers.

Finding Feature Information in This Module

Your software release might not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the "Feature Information for Monitoring and Maintaining Multilink Frame Relay" section on page 15-4.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Contents

It includes the following sections:

- Feature Overview, page 15-1
- Configuring Multilink Frame Relay, page 15-2
- Monitoring and Maintaining Frame Relay and Multilink Frame Relay, page 15-2

Feature Overview

The Multilink Frame Relay feature is based on the Frame Relay Forum's Multilink Frame Relay UNI Implementation Agreement (FRF.16). This feature provides a cost-effective way to increase bandwidth for certain applications by enabling multiple serial links to be aggregated into a single bundle of bandwidth. Multilink Frame Relay is supported on User-to-Network Interface (UNI) networks.

Configuring Multilink Frame Relay

The mulitlink frame relay configuration processes are described in the *Frame Relay - Multilink (MLFR-FRF.16)* feature guide. For more information, see the Frame Relay - Multilink (MLFR-FRF.16) feature guide at:
http://www.cisco.com/en/US/docs/ios/ios_xe/wan/configuration/guide/wan_ml_fr_frf161_xe.html

Monitoring and Maintaining Frame Relay and Multilink Frame Relay

You can use different **show** and **debug** commands to monitor and maintain frame relay and multilink frame relay configurations.

You can use the following **show** commands to display the multilink frame relay configurations:

- **show platform software frame-relay**—Displays the statistics about frame relay permanent virtual circuits (PVCs).
- **show platform software mfr**—Displays statistics about multilink frame relay information.

You can use the following **debug** commands to debug the multilink frame relay configurations:

- **debug platform hardware qfp active interface frame-relay multilink**—Debugs the multilink frame relay interfaces in the Cisco QuantumFlow Processor (QFP).
- **platform trace runtime process forwarding-manager module**—Enables Forwarding Manager Route Processor and Embedded Service Processor trace messages for the forwarding manager process.

For additional information on the commands and the options available for each command, see the *Cisco IOS Configuration Fundamentals Command Reference document,* located at:

http://www.cisco.com/en/US/docs/ios/fundamentals/command/reference/cf_book.html.

Additional References

Related Documents

Related Topic	Document Title
Multilink - Frame Relay (MLFR-FRF.16)	*http://www.cisco.com/en/US/docs/ios/ios_xe/wan/configuration/guide/wan_ml_fr_frf161_xe.html*
Frame Relay	*Wide-Area Networking Configuration Guide: Frame Relay, Cisco IOS XE Release 3S*
Cisco IOS commands	*Cisco IOS Master Commands List, All Releases*

Standards

Standard	Title
None	—

MIBs

MIB	MIBs Link
None	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at this URL: http://www.cisco.com/go/mibs

RFCs

RFC	Title
None	—

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for Monitoring and Maintaining Multilink Frame Relay

Table 15-1 lists the features in this module and provides links to specific configuration information.

Use Cisco Feature Navigator to find information about platform support and software image support. Cisco Feature Navigator enables you to determine which software images support a specific software release, feature set, or platform. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Note Table 15-1 lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Table 15-1 Feature Information for Monitoring and Maintaining Multilink Frame Relay

Feature Name	Releases	Feature Information
Monitoring and Maintaing Frame Relay	Cisco IOS XE 3.4.0S	In Cisco IOS XE Release 3.4S, this feature was introduced on the Cisco ASR 1000 Series Router.

Configuring MPLS Layer 2 VPNs

First Published: March 29, 2012
Lasted Revised: May 25, 2015

The Frame Relay to ATM Bridged Interworking feature provides interoperability between the Frame Relay attachment virtual circuit (VC) and the ATM attachment VC that are connected to different provider edge (PE) routers. The bridged encapsulation corresponding to the bridged (Ethernet) interworking mechanism is used to enable this interoperability. The Ethernet frames are carried through the MPLS network using Ethernet over MPLS (EoMPLS). The interworking function is performed in the PE routers connected to the Frame Relay attachment VC and the ATM attachment VC based on RFC 2684 and RFC 2427.

The xconnect support on Gigabit EtherChannel (GEC) Virtual Private Wire Service (VPWS) on ASR 1000 feature enables service providers to supply connectivity between customer sites with existing data link layer (Layer 2) networks by using a single, integrated, packet-based network infrastructure—a Cisco MPLS network. Instead of separate networks with separate network management environments, service providers can deliver Layer 2 connections over an MPLS backbone.

Layer 2 Gateway Protocol (L2GP) is a recommended IEEE standard (802.1ah) to address the issues that arise when two independent, bridged domains are connected redundantly through an arbitrary number of links. L2GP defines how the forwarding gateways are selected, so that only redundant ports are blocked and there are no temporary loops. The transition should be at least at the same speed in which Spanning Tree Protocol (STP) L2GP resolves the transient loop problem during reconvergence because it does not require cooperation from the outside domain.

Reverse Layer 2 Gateway Protocol (R-L2GP) is a variation of an L2GP. In case of an R-L2GP, the pseudo information of the R-L2GP is transmitted by network provider edges (nPEs) instead of user provider edges (uPEs). R-L2GP provides a mechanism to send out static preconfigured bridge protocol data units (BPDUs) on each ring access port of the nPEs to stimulate a per-access ring instantiation of the protocol. R-L2GP enables the PEs to avoid the burden of running Multiple Instances Spanning Tree Protocol (MISTP) when multiple independent access networks that run MISTP connect to a pair of redundant PEs.

High-Level Data Link Control (HDLC) Ethernet over MPLS is part of the Any Transport over MPLS (AToM) solution. HDLC and Ethernet are two link-layer transports that utilize the AToM architecture.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest information about features and caveats, see the release notes document pertaining to your platform and software release. To find information about the features documented in this module and to view a list of the

releases in which each feature is supported, see the "Feature Information for Configuring MPLS Layer 2 VPNs" section on page 16-70.

Use the Cisco Feature Navigator to find information about platform support and Cisco IOS and Cisco Catalyst operating system software image support. To access the Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Contents

- Overview of L2VPN Interworking, page 16-2
- Virtual Private LAN Services, page 16-4
- Frame Relay DLCI-to-ATM AAL5SNAP Bridged Interworking, page 16-13
- Configuring Frame Relay DLCI-to-ATM AAL5SNAP Bridged Interworking, page 16-15
- Gigabit EtherChannel for Virtual Private Wire Service, page 16-26
- Configuring Gigabit EtherChannel for Virtual Private Wire Service, page 16-28
- High-Level Data Link Control-Ethernet Interworking, page 16-39
- Configuring HDLC-Ethernet Interworking, page 16-40
- Additional References, page 16-69
- Feature Information for Configuring MPLS Layer 2 VPNs, page 16-70
- Glossary, page 16-73

Overview of L2VPN Interworking

Interworking is a transforming function that interconnects two heterogeneous attachment circuits (ACs). Several types of interworking functions exist. The function that is used depends on the AC type used, the type of data carried, and the level of functionality required. The two main Layer 2 Virtual Private Network (L2VPN) interworking functions supported in Cisco IOS XE software are bridged interworking and routed interworking.

Layer 2 (L2) transport over multiprotocol label switching (MPLS) and IP exists for ACs, such as Ethernet-to-Ethernet or Point-to-Point Protocol (PPP), Ethernet to VLAN, and Ethernet to Frame Relay. An interworking function facilitates translation between different L2 encapsulations.

L2VPN Interworking Modes

L2VPN interworking works in either Ethernet (bridged) mode or IP (routed) mode. You can specify the mode by issuing the **interworking {ethernet | ip}** command in pseudowire-class configuration mode and in L2VPN xconnect configuration mode for protocol-based CLI.

The **interworking** command causes the ACs to be terminated locally. The two keywords perform the following functions:

- The **ethernet** keyword causes Ethernet frames to be extracted from an AC and sent over the pseudowire. Ethernet end-to-end transmission is resumed. The AC frames that are not Ethernet are dropped. In the case of VLAN, the VLAN tag is removed, leaving an untagged Ethernet frame.

- The **ip** keyword causes IP packets to be extracted from an AC and sent over the pseudowire. The AC frames that do not contain IPv4 packets are dropped.

The following sections explain the Ethernet and IP interworking modes in detail.

Ethernet or Bridged Interworking

Ethernet interworking is also called bridged interworking. Ethernet frames are bridged across the pseudowire. The CE routers can natively bridge Ethernet traffic or can route traffic using a bridged encapsulation model, such as Bridge-group Virtual Interface (BVI) or Routed Bridge Encapsulation (RBE). The PE routers operate in the Ethernet like-to-like mode.

The Ethernet interworking mode offers the following services:

- LAN services—An example of this is an enterprise that has several sites, with some sites having Ethernet connectivity to the service provider (SP) network and others having Asynchronous Transfer Mode (ATM) connectivity. If the enterprise requires LAN connectivity to all its sites, traffic from the Ethernet or VLAN of one site can be sent through the IP/MPLS network and encapsulated as bridged traffic over an ATM VC of another site.
- Connectivity services—An example of this is an enterprise that has different sites running an Internal Gateway Protocol (IGP) that has incompatible procedures on broadcast and non broadcast links. This enterprise has several sites that run an IGP, such as Open Shortest Path First (OSPF) or Intermediate System-to-Intermediate System (IS-IS), between the sites. In this scenario, some of the procedures (such as route advertisement or designated router election) depend on the underlying L2 protocol and are different for a point-to-point ATM connection versus a broadcast Ethernet connection. Therefore, the bridged encapsulation over ATM can be used to achieve homogenous Ethernet connectivity between the CE routers running an IGP.

IP or Routed Interworking

IP interworking is also called routed interworking. The CE routers encapsulate the IP on the link between the CE router and the PE router. A new VC type is used to signal the IP pseudowire in MPLS. Translation between the L2 and IP encapsulations across the pseudowire is required. Special consideration needs to be given to the address resolution protocol operation and routing protocol operation, because these are handled differently on different L2 encapsulations.

The IP interworking mode is used to provide IP connectivity between sites, regardless of the L2 connectivity to these sites. It is different from a Layer 3 VPN because it is point-to-point in nature and the service provider does not maintain any routing information pertaining to customers.

Address resolution is encapsulation dependent as specified here:

- Ethernet uses Address Resolution Protocol (ARP)
- ATM uses inverse ARP
- PPP uses IP Control Protocol (IPCP)
- HDLC uses Serial Line ARP (SLARP)

Therefore, address resolution must be terminated on the PE router. Also, the end-to-end address resolution is not supported. Routing protocols operate differently over broadcast and point-to-point media. For Ethernet, the CE routers must either use static routing or configure the routing protocols to treat the Ethernet side as a point-to-point network.

In routed interworking, the IP packets that are extracted from the ACs are sent over the pseudowire. The pseudowire works in the IP Layer 2 transport (VC type 0x000B) like-to-like mode. The interworking function at the network service provider's (NSP) end completes the required adaptation based on the AC technology. The non-IPv4 packets are dropped.

In routed interworking, the following considerations must be kept in mind:

- ARP, inverse ARP, and IPCP are punted to the routing protocol.

  Therefore, the PE router at the NSP end must provide the following address-resolution functionalities for the *Ethernet* and *ATM and Frame Relay point-to-point subinterface* attachment circuits:

  - Ethernet—The PE device acts as a Proxy ARP server to all the ARP requests from the CE router. The PE router responds with the MAC address of its local interface.
  - ATM and Frame Relay point-to-point subinterface—By default, inverse ARP does not run in the point-to-point Frame Relay or ATM subinterfaces. The IP address and subnet mask define the connected prefix; therefore, configuration is not required in the CE devices.

- Interworking requires that the MTUs in both the ACs must match for the pseudowire that is to come up. The default MTU in one AC must match the MTU of other AC.

Table 16-1 lists the range of MTUs that can be configured for different ACs.

Table 16-1 Range of MTUs for Different ACs

AC Type	Range of MTUs[1] Supported
ATM	64 to 9216
Gigabit Ethernet	1500 to 9216
POS	64 to 9216
Fast Ethernet	1500 to 9216

1. The MTU configured on an AC must not exceed the MTU in the core network. This ensures that the traffic is not fragmented.

- The CE routers with Ethernet attachment VCs running OSPF must be configured with the *ospfIfType* option so that the OSPF protocol treats the underlying physical broadcast link as a P2P link.

Virtual Private LAN Services

Virtual Private LAN Service (VPLS) enables enterprises to link together their Ethernet-based LANs from multiple sites via the infrastructure provided by their service provider. From the enterprise perspective, the service provider's public network looks like one giant Ethernet LAN. For the service provider, VPLS provides an opportunity to deploy another revenue-generating service on top of their existing network without major capital expenditures. Operators can extend the operational life of equipment in their network.

Virtual Private LAN Services (VPLS) uses the provider core to join multiple attachment circuits together to simulate a virtual bridge that connects the multiple attachment circuits together. From a customer point of view, there is no topology for VPLS. All of the CE devices appear to connect to a logical bridge emulated by the provider core.

Reverse Layer 2 Gateway Protocol

Layer 2 Gateway Protocol (L2GP) is a recommended IEEE standard (802.1ah) to address the issues that arise when two independent, bridged domains are connected redundantly through an arbitrary number of links. L2GP defines how the forwarding gateways are selected, so that only redundant ports are blocked and there are no temporary loops. The transition should be at least the same speed in which STP L2GP resolves the the transient loop problem during the reconvergence because it does not require cooperation from the outside domain.

Reverse Layer 2 Gateway Protocol (R-L2GP) is a variation of an L2GP. In case of an R-L2GP, the pseudo information of the R-L2GP is transmitted by Network-facing Provider Edges (nPEs) instead of User Provider-Edges (uPEs). R-L2GP provides a mechanism to send out static preconfigured Bridge Protocol Data Units (BPDUs) on each ring access port of nPEs to stimulate a per-access ring instantiation of the protocol. R-L2GP enables the Provider Edges (PEs) to avoid the burden of running Multiple Instance Spanning Tree Protocol (MST) when multiple independent access networks that run MST connect to a pair of redundant PEs.

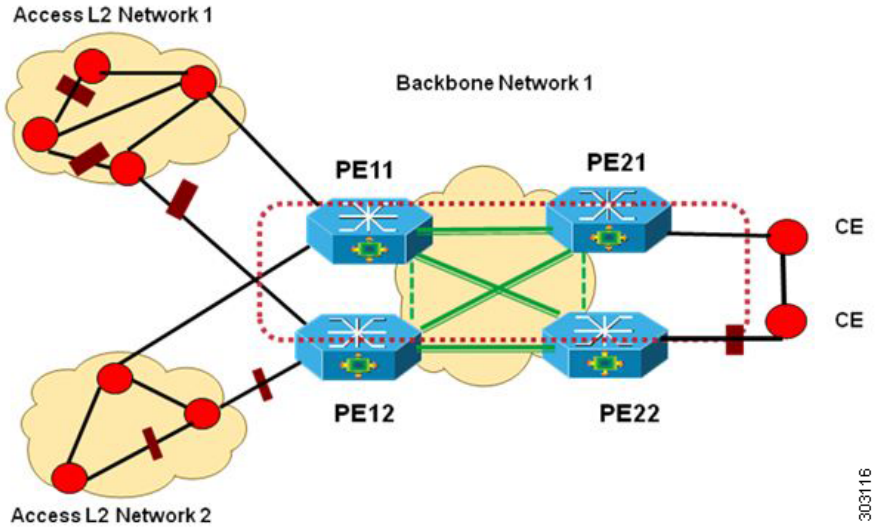
In order for this to work, the pair of nPEs are programmed to send out BPDUs on the access ring ports in such a way that they appear to be either:

- The root bridge itself (the bridge with the lowest bridge ID or priority).
- The bridge with the second lowest bridge ID or priority, and with a 0 cost path to the root.

Using R-L2GP, you can statically configure the BPDUs instead of the STP generate the BPDUs dynamically.

Figure 16-1 shows the topology of multiple-access networks connected to redundant nPEs.

Figure 16-1 Multiple-Access Networks Connected to Redundant nPEs



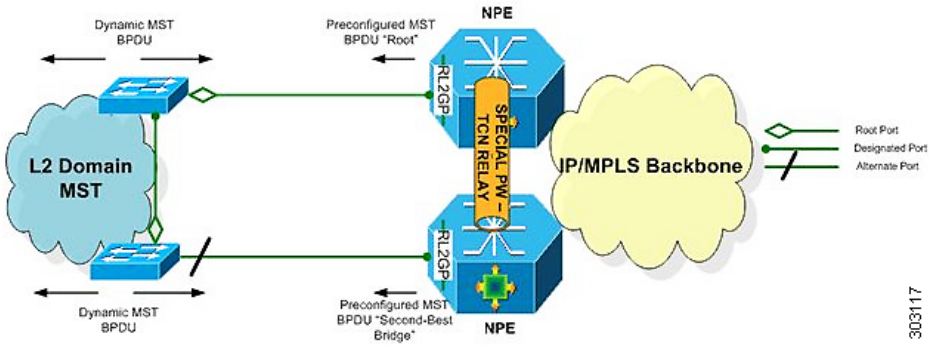
BPDUs Sent Out of R-L2GP Ports

An R-L2GP module in a route processor (RP) generates static preconfigured BPDUs, and sends them to uPEs via access ports, with the R-L2GP enabled.

Note Only localy generated static BPDUs can be sent out to RL2GP ports.

Figure 16-2 shows how a BPDU is forwarded to an R-L2GP port.

Figure 16-2 BPDU on an R-L2GP Port



BPDUs Received on R-L2GP Ports

On PE, only BPDUs with Topology Change Notification (TCN) bits on are punted to the R-L2GP and the STP module. If the PE is in a redundant setting, the corresponding BPDUs are is propagated to peer-redundant PE via the L2 protocol forwarding pseudowire (PW).

BPDUs Received on L2 Protocol Forwarding PW

The TCN BPDUs received from L2 protocol forwarding PW are punted to RP, and STP/R-L2GP process it and generate MAC flush.

Restrictions for R-L2GP

The restrictions for the R-L2GP feature are:

- R-L2GP is supported only on L2 bridge ports, and is not compatible with prestandard MST.
- All the access-side shall have the same MST instance, the same name and the same revision number configuration as nPEs.
- There is no configure error detection and recover mechanism for R-L2GP. Users are expected to configure R-L2GP and MSTP instance on CEs and nPEs correctly.

Configuring the R-L2GP

Since the R-L2GP configuration is bundled with the MST configuration, the above parameters can be recycled from the MSTI and MST region (currently only one MST region is supported on IOS) configurations. This section describes how to configure Reverse L2GP. It consists of the following sections:

- Configuring the MST, page 16-8
- Configuring an R-L2GP Instance, page 16-9
- Attaching an R-L2GP Instance to a Port, page 16-10
- Example: Configuring an R-L2GP, page 16-11

- Configuring the Layer 2 Protocol Forwarding Virtual Private LAN Services Pseudowire Between Two Redundant NPES, page 16-11

Configuring the MST

Configuration of the MST must be done before configuring the R-L2GP and attaching the R-L2GP to a port.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree mode mst**
4. **spanning-tree mst configuration**
5. [**no**] **name** *name*
6. [**no**] **revision** *version*
7. [**no**] **instance** *instance-id* {**vlans** *vlan-range*}

DETAILED STEPS

	Command	Purpose
Step 1	**enable** **Example:** `Router# enable`	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**spanning-tree mode mst** **Example:** `Router(config)# spanning-tree mode mst.`	Enables MST mode.
Step 4	**spanning-tree mst configuration** **Example:** `Router(config)# spanning-tree mst configuration`	Enters MST configuration submode.
Step 5	**name** *name* **Example:** `Router(config-mst)# name Cisco`	Sets the name of the MST region. **Note** All the nodes in the same region should be configured with the same MST name.

	Command	Purpose
Step 6	**revision** *version* **Example:** `Router(config-mst)# revision 5`	Sets the revision number for the MST (802.1s) configuration. **Note** All the nodes in the same region should be configured with the same MST configure revision number.
Step 7	**instance** *instance-id* {**vlans** *vlan-range*} **Example:** `Router(config-mst)# instance 2 vlans 1-100`	Maps a VLAN or a group of VLANs to an MST instance.

Configuring an R-L2GP Instance

Perform the following steps to configure R-L2GP instance.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree pseudo-information transmit** *indentifier*
4. **remote-id** *id*
5. **mst root**
6. **mst cost**

DETAILED STEPS

	Command	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. Enter your password, if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**spanning-tree pseudo-information transmit** *indentifier* **Example:** `Router(config)# spanning-tree pseudo-information transmit 46`	Configures the Reverse-L2GP configuration on the interface or the untagged Ethernet Flow Point (EFP) port.
Step 4	**remote-id** *id* **Example:** `Router(config-pseudo)# remote-id 53`	Configures the remote R-L2GP instance ID that pairs with the specified R-L2GP instance ID.

	Command	Purpose
Step 5	**mst** *region-id* **root** *mac-address* **Example:** `Router(config-pseudo)# mst 0 root 32768 0000.0000.0001`	Adds MST instances to R-L2GP instances and configures the MAC address and priority for MST instances. **Note** MST 0 has all the VLANs that have not been explicitly specified in other MST instances. MST 0 must be configured for each R-L2GP instance.
Step 6	**mst** *region-id* **cost** **Example:** `Router(config-pseudo)# mst 1 cost 1`	Adds the corresponding MST instance list to the R-L2GP instance and configures the R-L2GP path cost for the MST instance or multiple MST instances.

Note To configure an R-L2GP on the Cisco ASR 1000 Series Aggregation Services Router, the *remote-id* configured on nPE1 must be the *transmit identifier* configured on nPE2, and vice versa, as show in *Example: Configuring an R-L2GP*.

Attaching an R-L2GP Instance to a Port

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface gigabitethernet** *slot/port* or **interface tengigabitethernet** *slot/port*
4. **spanning-tree pseudo-information transmit** *indentifier*

DETAILED STEPS

	Command	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. Enter your password, if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.

	Command	Purpose
Step 3	**interface gigabitethernet** *slot/port* or **interface tengigabitethernet** *slot/port* **Example:** `Router(config)# interface gigabitethernet 4/1`	Specifies Gigabit Ethernet or the 10 Gigabit Ethernet interface on the access side of the nPE to configure. Here: • *slot/port*—Specifies the location of the interface.
Step 4	**spanning-tree pseudo-information transmit** *identifier* **Example:** `Router(config-if)# spanning-tree pseudo-information transmit 46`	Configures the Reverse-L2GP configuration on the interface. **Note** The identifier should be the same as the one configured on the nPE.

Example: Configuring an R-L2GP

The following example shows how to configure an R-L2GP in a network comprising two nPEs.

Configuration example on nPE1:

```
enable
configure terminal
spanning-tree pseudo-information transmit 46
  remote-id 53
  mst 0 root 32768 0000.0000.0001
  mst 1 root 32768 0000.0000.0002
  mst 1 cost 1
  mst 2 root 32768 0000.0000.0003
exit
interface gigabitEthernet 2/1/0
   spanning-tree pseudo-information transmit 46
```

Configuration example on nPE2:

```
spanning-tree pseudo-information transmit 53
  remote-id 46
  mst 0 root 32768 0000.0000.0001
  mst 1 root 32768 0000.0000.0002
  mst 1 cost 1
  mst 2 root 32768 0000.0000.0003
interface gigabitEthernet 0/0/1
   spanning-tree pseudo-information transmit 53
```

Configuring the Layer 2 Protocol Forwarding Virtual Private LAN Services Pseudowire Between Two Redundant NPES

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **l2 vfi** *vfi-name* **manual**
4. **vpn id** *vpn_id*
5. **bridge-domain** *bridge_id*

6. **forward permit l2protocol all**
7. **neighbor** *ip-address vc-id* {**encapsulation mpls** |**pw-class** *pw-class-name*}

DETAILED STEPS

	Command	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. Enter your password, if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**l2 vfi** *vfi-name* **manual** **Example:** `Router(config)# l2 vfi vfitest1 manual`	Creates a Layer 2 Virtual Forwarding Instance (VFI) and enters the Layer 2 VFI manual configuration submode.
Step 4	**vpn id** *vpn_id* **Example:** `Router(config-vfi)# vpn id 303`	Sets or updates a VPN ID on a VPN routing and forwarding (VRF) instance.
Step 5	**bridge-domain** bridge_id **Example:** `Router(config-vfi)# bridge-domain 100`	Binds a service instance to a bridge domain instance.
Step 6	**forward permit l2protocol all** **Example:** `Router(config-vfi)# forward permit l2protocol all`	Defines the VPLS pseudowire that is used to transport bridge protocol data unit (BPDU) information between two network provider edge (N-PE) routers.
Step 7	**neighbor** *ip-address vc-id* {**encapsulation mpls** |**pw-class** *pw-class-name*} **Example:** `Router(config-vfi)# neighbor 10.10.10.10 1 encapsulation mpls`	Specifies the routers that should form a point-to-point Layer 2 virtual forwarding interface (VFI) connection.

Verifying an R-L2GP Configuration

The following examples show how to use the **show** commands to verify an R-L2GP configuration:

```
Router# show spanning-tree pseudo-information 46 configuration

remote_id 53
 mst_region_id 0, port_count 2, update_flag 0x0
 mrecord 0x3AF841EC, mrec_count 3:
      msti 0: root_id 32768.0000.0000.0001, root_cost 0, update_flag 0x0
      msti 1: root_id 32769.0000.0000.0002, root_cost 1, update_flag 0x0
```

```
        msti 2: root_id 32770.0000.0000.0003, root_cost 0, update_flag 0x0

Router# show spanning-tree pseudo-information 1 interface GigabitEthernet3/0/3

Pseudo id 1:
        GigabitEthernet 2/1/0
        GigabitEthernet 0/0/1
```

Prerequisites for Frame Relay DLCI-to-ATM AAL5SNAP Bridged Interworking

Before you configure the Frame Relay Data Link Connection Identifier (DLCI)-to-ATM AAL5SNAP Bridged Interworking feature on a router, ensure that the following prerequisites are met:

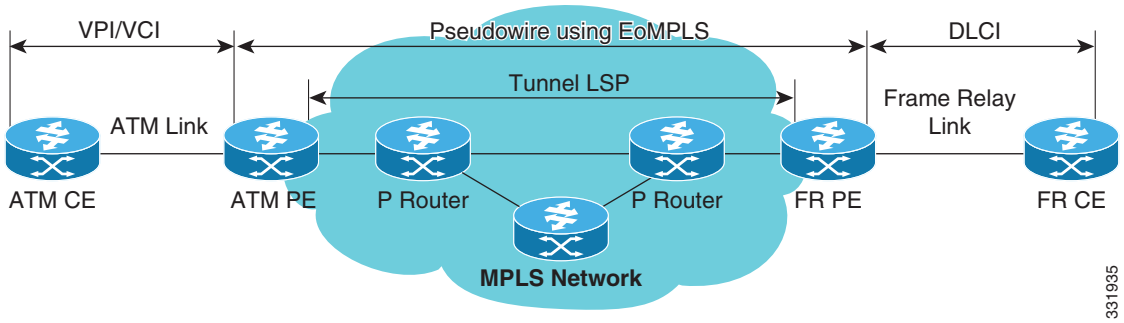
- Enable frame-relay switching on the Frame Relay provider edge (PE) router.
- Customer edge (CE) routers must support Bridge-group Virtual Interface or Routed Bridge Encapsulation.

Frame Relay DLCI-to-ATM AAL5SNAP Bridged Interworking

This feature provides interoperability between the ATM attachment VC and Frame Relay attachment VC connected to different PE routers. This interworking uses the bridged encapsulation corresponding to the bridged (Ethernet) interworking mechanism. The Ethernet frames are carried through the MPLS network using Ethernet over MPLS (EoMPLS). This feature is configured only in the bridged mode and not in the routed mode.

Figure 16-3 shows the interworking function performed in the PE routers that are connected to the ATM attachment VC and the Frame Relay attachment VC.

Figure 16-3 Network Topology for Frame Relay DLCI-to-ATM AAL5SNAP Bridged Interworking



On the ATM PE router with interworking function, when traffic flows from the ATM segment to MPLS cloud, the bridged encapsulation (ATM and SNAP header) is discarded and the Ethernet frame is encapsulated with the labels required to go through the pseudowire using the VC type 5 (Ethernet). In the opposite direction, after the label disposition from the MPLS cloud, the Ethernet frames are encapsulated over AAL5SNAP using bridged encapsulation.

On the FR PE router with interworking function, when traffic flows from the FR segment to the MPLS cloud, the bridged encapsulation (Frame Relay and SNAP header) is discarded and the Ethernet frame is encapsulated with the labels required to go through the pseudowire, using the VC type 5 (Ethernet). In the opposite direction, after the label disposition from the MPLS cloud, the Ethernet frames are encapsulated over FR using bridged encapsulation.

The PE router automatically supports translation of both Cisco and IETF Frame Relay encapsulation types coming from the Customer edge (CE) router, but translates only to IETF when sending to the CE router. The Cisco CE router can handle the IETF encapsulation on receipt, even if it is configured to send Cisco encapsulation.

The following modes are supported:

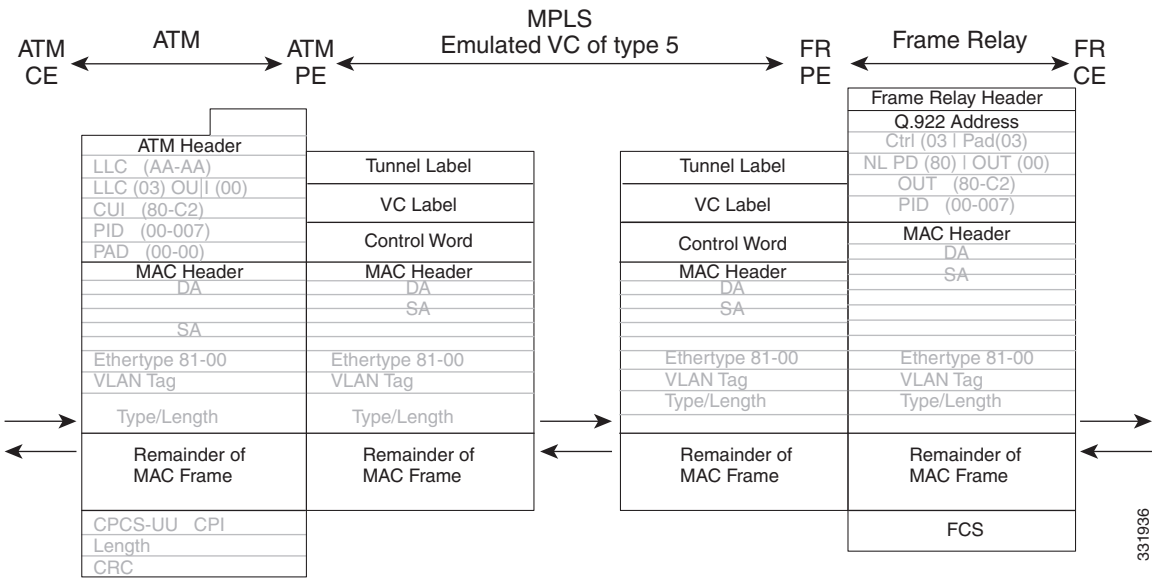
- The ATM permanent virtual circuit (PVC) mode with the AAL5SNAP encapsulation type, and the existing Quality of Service (QoS) functionality for ATM PVCs.
- The Frame Relay DLCI mode, and the existing QoS functionality for Frame Relay.

PVC status signaling works the same way it does in the like-to-like case. The PE router reports the PVC status to the CE router, based on the availability of the pseudowire.

The attachment circuit maximum transmission unit (MTU) on both sides of the pseudowire must match when connected over MPLS. The non-AAL5 traffic (such as OAM cells) is punted to be processed at the RP level. A VC that is configured with OAM cell emulation on the ATM PE router (using the **oam-ac emulation-enable** command) can send end-to-end F5 loopback cells at configured intervals toward the CE router. When the pseudowire is down, an end-to-end F5 segment alarm indication signal (AIS) and remote defect indication (RDI) is sent from the PE router to the CE router.

Figure 16-4 shows the protocol stack for the Frame Relay DLCI-to-ATM AAL5SNAP Bridged Interworking feature.

Figure 16-4 Protocol Stack for Frame Relay DLCI-to-ATM AAL5SNAP Bridged Interworking



Configuring Frame Relay DLCI-to-ATM AAL5SNAP Bridged Interworking

To configure the Frame Relay DLCI-to-ATM AAL5SNAP Bridged Interworking feature on an ATM-PE router, perform the following steps:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **no ip domain lookup**
4. **mpls label range** *minimum-value maximum-value* [**static** *minimum-static-value maximum-static-value*]
5. **mpls label protocol ldp**
6. **mpls ip default-route**
7. **mpls ldp graceful-restart**
8. **xconnect logging pseudowire status**
9. **pseudowire-class** [*pw-class-name*]
10. **encapsulation mpls**
11. **interworking ethernet**
12. **exit**
13. **interface loopback** *loopback-interface-number*
14. **ip address** *ip-address mask*
15. **exit**
16. **interface GigabitEthernet** *slot/subslot/port*
17. **ip address** *ip-address mask*
18. **negotiation auto**
19. **mpls ip**
20. **exit**
21. **interface atm** *slot/subslot/port*
22. **no ip address**
23. **atm clock internal**
24. **no atm enable-ilmi-trap**
25. **exit**
26. **interface atm** *slot/subslot/port* [*.subinterface-number* {**point-to-point**}]
27. **mtu** *bytes*
28. **no atm enable-ilmi-trap**
29. **pvc** [*name*] *vpi/vci* **l2transport**
30. **encapsulation** *encapsulation-type*

31. **xconnect** *peer-ip-address vc-id* **encapsulation mpls pw-class** *pw-class-name*
32. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables the privileged EXEC mode. Enter your password, if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters the global configuration mode.
Step 3	**Router(config)# no ip domain lookup**	Disables the IP domain naming system (DNS).
Step 4	**mpls label range** *minimum-value maximum-value* **[static** *minimum-static-value maximum-static-value***]** **Example:** `Router(config)# mpls label range 101 4000 static 4001 5001`	Configures the range of local labels available for use with Multiprotocol Label Switching (MPLS) applications on packet interfaces.
Step 5	**mpls label protocol ldp** **Example:** `Router(config)# mpls label protocol ldp`	Specifies label distribution protocol (LDP) for the ATM-PE router.
Step 6	**mpls ip default-route** **Example:** `Router(config)# mpls ip default-route`	Enables the distribution of labels associated with the IP default route.
Step 7	**mpls ldp graceful-restart** **Example:** `Router(config)# mpls ldp graceful-restart`	Enables MPLS LDP graceful restart.
Step 8	**xconnect logging pseudowire status** **Example:** `Router(config)# xconnect logging pseudowire status`	Enables system logging (syslog) reporting of pseudowire status events.
Step 9	**pseudowire-class [***pw-class-name***]** **Example:** `Router(config)# pseudowire-class atm-fr-bridged`	Establishes a pseudowire class with a name that you specify, and enters the pseudowire class configuration mode.
Step 10	**encapsulation mpls** **Example:** `Router(config-pw-class)# encapsulation mpls`	Enables MPLS encapsulation on the interface.
Step 11	**interworking ethernet** **Example:** `Router(config-pw-class)# interworking ethernet`	Enables the L2VPN Ethernet interworking feature.

	Command or Action	Purpose
Step 12	**exit**	Exits pseudowire class configuration mode.
Step 13	**interface loopback** *loopback-interface-number* **Example:** `Router(config)# interface loopback 0`	Specifies the loopback logical interface.
Step 14	**ip address** *ip-address mask* **Example:** `Router(config-if)# ip address 44.1.1.2 255.255.255.255`	Specifies the IP address for the Loopback interface.
Step 15	**exit**	Exits interface configuration mode.
Step 16	**interface GigabitEthernet** *slot/subslot/port* **Example:** `Router(config)# interface GigabitEthernet 0/0/1`	Specifies the Gigabit Ethernet interface for the connection of the PE routers.
Step 17	**ip address** *ip-address mask* **Example:** `Router(config-if)# ip address 10.10.1.2 255.255.255.0`	Specifies the IP address for the Gigabit Ethernet interface.
Step 18	**negotiation auto** **Example:** `Router(config-if)# negotiation auto`	Enables the auto negotiation protocol to configure the speed, duplex, and automatic flow control of the Gigabit Ethernet interface.
Step 19	**mpls ip** **Example:** `Router(config-if)# mpls ip`	Enables MPLS forwarding of the IPv4 packets towards the MPLS core.
Step 20	**exit**	Exits interface configuration mode.
Step 21	**interface atm** *slot/subslot/port* **Example:** `Router(config)# interface atm 0/1/2`	Configures an ATM interface and enters interface configuration mode.
Step 22	**no ip address** **Example:** `Router(config-if)# no ip address`	Removes the previously configured IP address.
Step 23	**atm clock internal** **Example:** `Router(config-if)# atm clock internal`	Enables the ATM interface to generate the transmit clock internally.

	Command or Action	Purpose
Step 24	**no atm enable-ilmi-trap** **Example:** `Router(config-if)# no atm enable-ilmi-trap`	Disables the Integrated Local Management Interface (ILMI) ATM traps.
Step 25	**exit**	Exits interface configuration mode.
Step 26	**interface atm** *slot/subslot/port* **[**.*subinterface-number* **{point-to-point}]** **Example:** `Router(config)# interface atm 0/1/2.1 point-to-point`	Configures an ATM interface and enters interface configuration mode.
Step 27	**mtu** *bytes* **Example:** `Router(config-subif)# mtu 1500`	Adjusts the maximum packet size or maximum transmission unit (MTU) size. **Note** The MTU sizes of both the attachment circuits must match.
Step 28	**no atm enable-ilmi-trap** **Example:** `Router(config-subif)# no atm enable-ilmi-trap`	Disables the ILMI ATM traps.
Step 29	**pvc [***name***]** *vpi/vci* **l2transport** **Example:** `Router(config-subif)# pvc cisco 10/100 l2transport`	Assigns a name to an ATM PVC, specifies the encapsulation type on an ATM PVC, and enters ATM virtual circuit configuration mode.
Step 30	**encapsulation** *encapsulation-type* **Example:** `Router(config-if-atm-l2trans-pvc)# encapsulation aal5snap`	Sets the AAL5SNAP encapsulation (Any-to-Any) for the ATM point-to-point interface.
Step 31	**xconnect** *peer-ip-address vc-id* **encapsulation mpls pw-class** *pw-class-name* **Example:** `Router(config-if-atm-l2trans-pvc)# xconnect 190.1.1.1 100 encapsulation mpls pw-class atm-fr-bridged`	Binds an attachment circuit to a pseudowire and configures an Any Transport over MPLS (AToM) static pseudowire.
Step 32	**exit**	Exits global configuration mode.

Example: Frame Relay-to-ATM Bridged Interworking on an ATM-PE Router

The following example shows the configuration of the Frame Relay-to-ATM Bridged Interworking feature on an ATM-PE router:

```
no ip domain lookup
mpls label range 101 4000 static 4001 5001
mpls label protocol ldp
mpls ip default-route
mpls ldp graceful-restart
xconnect logging pseudowire status
!
```

```
pseudowire-class atm-fr-bridged
    encapsulation mpls
    interworking ethernet
!
interface Loopback0
    ip address 44.1.1.2 255.255.255.255
!
interface GigabitEthernet0/0/1
    ip address 10.10.1.2 255.255.255.0
    negotiation auto
    mpls ip
!
interface ATM0/1/2
    no ip address
    atm clock INTERNAL
    no atm enable-ilmi-trap
!
interface ATM0/1/2.1 point-to-point
    mtu 1500
    no atm enable-ilmi-trap
    pvc 10/100 l2transport
        encapsulation aal5snap
        xconnect 190.1.1.1 100 pw-class atm-fr-bridged
    !
!
```

To configure the Frame Relay-to-ATM Bridged Interworking feature on a Frame Relay PE router, perform the following steps:

Note The following configuration uses a channelized T1/E1 interface. Frame Relay can be configured on other interfaces such as Packet over SONET (PoS) as well.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **(Optional) ipv6 unicast-routing**
4. **mpls label protocol ldp**
5. **mpls ip default-route**
6. **mpls ldp graceful-restart**
7. **frame-relay switching**
8. **xconnect logging pseudowire status**
9. **controller t1** *slot/subslot/port*
10. **framing esf**
11. **clock source internal**
12. **linecode b8zs**
13. **cablelength long** *db-loss-value*
14. **channel-group** *channel-group-number* **timeslots** ***range***
15. **exit**

16. **pseudowire-class** [*pw-class-name*]
17. **encapsulation mpls**
18. **interworking ethernet**
19. **exit**
20. **interface loopback** *loopback-interface-number*
21. **ip address** *ip-address mask*
22. **exit**
23. **interface serial** *slot/subslot/port:timeslot*
24. **no ip address**
25. **encapsulation frame-relay**
26. **frame-relay intf-type dce**
27. **frame-relay interface-dlci** *dlci* **switched**
28. **exit**
29. **interface GigabitEthernet** *slot/subslot/port*
30. **ip address** *ip-address mask*
31. **negotiation auto**
32. **mpls ip**
33. **exit**
34. **connect** *connection-name interface dlci* **l2transport**
35. **xconnect** *peer-ip-address vc-id* **encapsulation mpls pw-class** *pw-class-name*
36. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. Enter your password, if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**ipv6 unicast-routing** **Example:** `Router# ipv6 unicast-routing`	(Optional) Enables the task of forwarding the IPv6 unicast datagrams.
Step 4	**mpls label protocol ldp** **Example:** `Router(config)# mpls label protocol ldp`	Specifies the label distribution protocol (LDP) for the Frame Relay PE router.
Step 5	**mpls ip default-route** **Example:** `Router(config)# mpls ip default-route`	Enables the distribution of labels associated with the IP default route.
Step 6	**mpls ldp graceful-restart** **Example:** `Router(config)# mpls ldp graceful-restart`	Enables MPLS LDP graceful restart.
Step 7	**frame-relay switching** **Example:** `Router(config)# frame-relay switching`	Enables PVC switching on a Frame Relay data circuit-terminating equipment (DCE) device.
Step 8	**xconnect logging pseudowire status** **Example:** `Router(config)# xconnect logging pseudowire status`	Enables system logging (syslog) reporting of pseudowire status events.
Step 9	**controller t1 slot/subslot/port** **Example:** `Router(config)# controller T1 0/3/0`	Configures a T1 controller and enters the controller configuration mode.
Step 10	**framing esf** **Example:** `Router(config-controller)# framing esf`	Selects the Extended Super Frame (ESF) for a T1 data line.

	Command or Action	Purpose
Step 11	**clock source internal** **Example:** `Router(config-controller)# clock source internal`	Configures the clock source of a DS1 link and uses the internal clock from the interface.
Step 12	**linecode b8zs** **Example:** `Router(config-controller)# linecode b8zs`	Specifies Binary 8-Zero Substitution (B8ZS) as the line code type for the T1 controller.
Step 13	**cablelength long db-loss-value** **Example:** `Router(config-controller)# cablelength long 0db`	Decreases the transmit signal by 0 dB. This is the default value.
Step 14	**channel-group** ***channel-group-number*** **timeslots** ***range*** **Example:** **Router(config-controller)# channel-group 0 timeslots 1-24**	Configures serial WAN on a T1 or E1 interface.
Step 15	**exit**	Exits pseudowire class configuration mode.
Step 16	**pseudowire-class** [*pw-class-name*] **Example:** `Router(config)# pseudowire-class atm-fr-bridged`	Establishes a pseudowire class with a name that you specify and enters the pseudowire class configuration mode.
Step 17	**encapsulation mpls** **Example:** `Router(config-pw-class)# encapsulation mpls`	Enables MPLS encapsulation on the interface.
Step 18	**interworking ethernet** **Example:** `Router(config-pw-class)# interworking ethernet`	Enables the L2VPN Ethernet interworking feature.
Step 19	**exit**	Exits pseudowire class configuration mode.
Step 20	**interface loopback** *loopback-interface-number* **Example:** `Router(config)# interface loopback 0`	Specifies the Loopback logical interface.
Step 21	**ip address** *ip-address mask* **Example:** `Router(config-if)# ip address 44.1.1.2 255.255.255.255`	Specifies the IP address for the Loopback interface.
Step 22	**exit**	Exits interface configuration mode.

	Command or Action	Purpose
Step 23	**interface serial** slot/subslot/port:timeslot **Example:** Router(config)# **interface Serial**0/3/0:0	Specifies a serial interface created on a channelized T1 controller.
Step 24	**no ip address** **Example:** Router(config-if)# **no ip address**	Removes the previously configured IP address.
Step 25	**encapsulation frame-relay** **Example:** Router(config-if)# **encapsulation frame-relay**	Configures the Cisco Frame Relay encapsulation on serial interface.
Step 26	**frame-relay intf-type dce** **Example:** Router(config-if)# **frame-relay intf-type dce**	Configures a Frame Relay switch type. The router functions as a switch connected to a router.
Step 27	**frame-relay interface-dlci** *dlci* **switched** **Example:** Router(config-if)# **frame-relay interface-dlci** 101 **switched**	Assigns a data-link connection identifier (DLCI) to a specified Frame Relay subinterface on the router.
Step 28	**exit**	Exits interface configuration mode.
Step 29	**interface GigabitEthernet** *slot/subslot/port* **Example:** Router(config)# **interface GigabitEthernet** 0/0/1	Specifies the Gigabit Ethernet interface for the connection of the PE routers.
Step 30	**ip address** *ip-address mask* **Example:** Router(config-if)# **ip address** 10.10.1.2 255.255.255.0	Specifies the IP address for the Gigabit Ethernet interface.
Step 31	**negotiation auto** **Example:** Router(config-if)# **negotiation auto**	Enables the auto negotiation protocol to configure the speed, duplex, and automatic flow control of the Gigabit Ethernet interface.
Step 32	**mpls ip** **Example:** Router(config-if)# **mpls ip**	Enables MPLS forwarding of the IPv4 packets towards the MPLS core.
Step 33	**exit**	Exits interface configuration mode.

	Command or Action	Purpose
Step 34	**connect** *connection-name interface dlci* **l2transport** **Example:** `Router(config)# connect fr-atm-2 Serial0/3/0:0 101 l2transport`	Defines connections between the Frame Relay PVCs.
Step 35	**xconnect** ***peer-ip-address vc-id*** **encapsulation mpls pw-class** *pw-class-name* **Example:** `Router(config-conn)# xconnect 190.1.1.1 100 encapsulation mpls pw-class atm-fr-bridged`	Binds an attachment circuit to a pseudowire and configures an AToM static pseudowire.
Step 36	**exit**	Exits the global configuration mode.

Example: Frame Relay-to-ATM Bridged Interworking on a Frame Relay-PE Router

The following example shows the configuration of the Frame Relay-to-ATM Bridged Interworking feature on a Frame Relay-PE router:

```
ipv6 unicast-routing
mpls label protocol ldp
mpls ip default-route
mpls ldp graceful-restart
frame-relay switching
xconnect logging pseudowire status
!
controller T1 0/3/0
    framing esf
    clock source internal
    linecode b8zs
    cablelength long 0db
    channel-group 0 timeslots 1-24
!
pseudowire-class atm-fr-bridged
    encapsulation mpls
    interworking ethernet
!
interface Loopback0
    ip address 190.1.1.1 255.255.255.255
!
interface Serial0/3/0:0
    no ip address
    encapsulation frame-relay
    frame-relay intf-type dce
    frame-relay interface-dlci 101 switched
!
interface GigabitEthernet1/3/1
    ip address 10.10.1.1 255.255.255.0
    negotiation auto
    mpls ip
!

connect fr-atm-2 Serial0/3/0:0 101 l2transport
    xconnect 44.1.1.2 100 pw-class atm-fr-bridged
!
```

Gigabit EtherChannel for Virtual Private Wire Service

GEC for AToM is a solution for a VPWS transporting Layer 2 packets over an MPLS backbone with GEC.

This feature enables service providers to supply connectivity between customer sites having data link layer (Layer 2) networks, by using a single, integrated, packet-based network infrastructure—a Cisco MPLS network. Instead of separate networks with separate network management environments, service providers can deliver Layer 2 connections over an MPLS backbone.

Supported Modes

The following modes are supported in the GEC for VPWS feature:

- GEC Like-to-Like Mode, page 16-26
- Any-to-GEC Mode, page 16-27

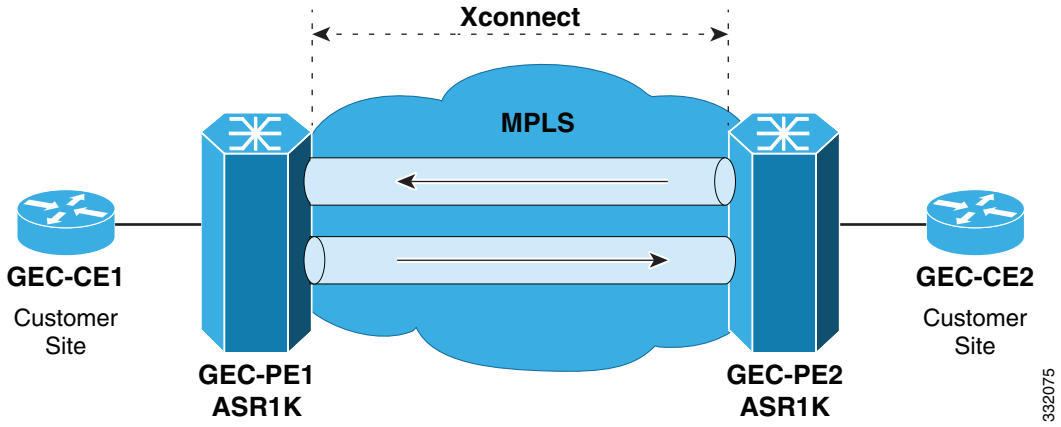
GEC Like-to-Like Mode

The GEC Like-to-Like mode allows switching of data between two physical interfaces in which the two segments (CE1-PE1 and CE2-PE2, as shown in Figure 16-5) are both of GEC type.

The GEC Like-to-Like mode has the following features:

- EtherChannel-to-EtherChannel over MPLS (Bridged) Interworking, page 16-28
- EtherChannel-to-EtherChannel over MPLS (Routed) Interworking, page 16-30

Figure 16-5 Topology of the GEC Like-to-Like Mode for the GEC for VPWS Feature



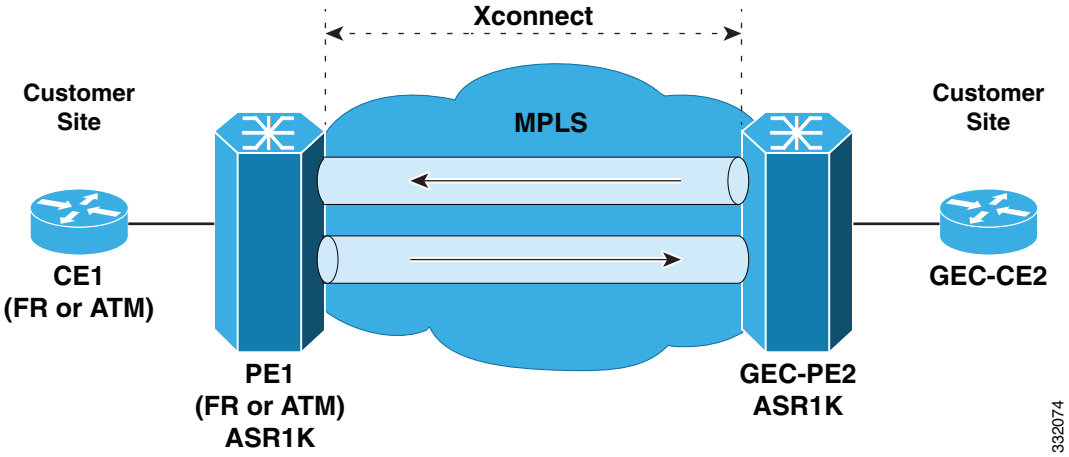
Any-to-GEC Mode

The Any-to-GEC mode allows switching of data between two physical interfaces in which the two segments, CE1-PE1 and CE2-PE2, are both of different types, while one is GEC, the other can be PPP, Ethernet, Frame Relay, or ATM, as shown in Figure 16-6.

The Any-to-GEC mode has the following features:

- Any-to-EtherChannel over MPLS (Bridged) Interworking, page 16-33
- Any-to-EtherChannel over MPLS (Routed) Interworking, page 16-35

Figure 16-6 Topology of the Any-to-GEC Mode for the GEC for VPWS Feature



Note Bridged interworking is used when Layer 2 (L2) packets are considered without regard for Layer 3 contents. In bridged interworking, Ethernet frames that are extracted from the attachment circuit are sent over the MPLS pseudowire.

Note Routed interworking is used to carry Layer 3 packets. In routed interworking, IP packets that are extracted from the attachment circuits are sent over the MPLS pseudowire.

Restrictions for Gigabit EtherChannel for Virtual Private Wire Service

The following are the restrictions for Gigabit EtherChannel for VPWS are the followings:

- GEC for VPWS does not support Q-in-Q encapsulation and remote port shutdown.
- A maximum four member links are supported under the port channel and a maximum of 64 port channel bundles are supported per router.

Configuring Gigabit EtherChannel for Virtual Private Wire Service

The GEC VPWS support feature is supported by AToM on the EtherChannel Interface, and includes the following features:

- EtherChannel-to-EtherChannel over MPLS (Bridged) Interworking, page 16-28
- EtherChannel-to-EtherChannel over MPLS (Routed) Interworking, page 16-30
- Any-to-EtherChannel over MPLS (Bridged) Interworking, page 16-33
- Any-to-EtherChannel over MPLS (Routed) Interworking, page 16-35

EtherChannel-to-EtherChannel over MPLS (Bridged) Interworking

Configure L2VPN interworking on the upstream interfaces of the PE routers. For more information about configuring L2VPN interworking on the PE routers, see L2VPN Interworking Modes, page 16-2.

After configuring MPLS Forwarding, perform the following steps on the downstream interfaces of the PE routers:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **mpls label protocol ldp**
4. **interface loopback** *loopback-interface-number*
5. **ip address** *ip-address ip-subnet-mask*
6. **exit**
7. **pseudowire-class** *pw-class-name*
8. **encapsulation mpls**
9. **interworking ethernet**
10. **exit**
11. **interface port-channel** *number*
12. **xconnect** *peer-ip-address vc-id* **encapsulation mpls pseudowire-class** *pw-class-name*
13. **interface GigabitEthernet** *slot* | *subslot* | *port*
14. **channel-group** *port-channel number*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable**	Changes the privilege level for the corresponding CLI session.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**mpls label protocol ldp** **Example:** `Router# mpls label protocol ldp`	Specifies that LDP is the default label distribution protocol.
Step 4	**interface loopback** *loopback-interface-number* **Example:** `Router# interface loopback 1`	Specifies the loopback interface, and enters interface configuration mode.
Step 5	**ip address** *ip-address mask* **Example:** `Router# ip address 10.10.2.1 255.255.255.0`	Sets the IP address and mask for the loopback interface.
Step 6	**exit**	Exits interface configuration mode.
Step 7	**pseudowire-class** *pw-class-name* **Example:** `Router(config)# pseudowire-class gec-bridged`	Specifies the name of a Layer 2 pseudowire class and enters the pseudowire class configuration mode.
Step 8	**encapsulation mpls** **Example:** `Router(config-pw)# encapsulation mpls`	Uses MPLS as the tunneling method to encapsulate data in the pseudowire.
Step 9	**interworking ethernet** **Example:** `Router(config-pw)# interworking ethernet`	Enables the L2VPN Interworking feature, and causes Ethernet frames to be extracted from the attachment circuit and sent over the pseudowire. Ethernet end-to-end transmission is assumed. Attachment circuit frames that do not contain Ethernet frames are dropped. In the case of VLAN, the VLAN tag is removed, which leaves a pure Ethernet frame.
Step 10	**exit**	Exits xconnect configuration mode.
Step 11	**interface port-channel** *number* **Example:** `Router(config)# interface port-channel 1`	Creates an EtherChannel interface on the Cisco Cable Modem Termination System (CMTS).

	Command or Action	Purpose
Step 12	**xconnect** *peer-ip-address vc-id* **encapsulation mpls pseudowire-class** *pw-class-name* **Example:** `Router(config-if)# xconnect 10.0.0.1 707 encapsulation mpl pseudowire-class gec-bridged`	Binds an attachment circuit to a pseudowire to configure an AToM static pseudowire, specifies MPLS as the tunneling method, and enters the xconnect configuration mode.
Step 13	**interface GigabitEthernet** *slot* | *subslot* | *port* **Example:** `Router(config)# interface GigabitEthernet 0/0/1`	Specifies the Gigabit Ethernet interface, and enters interface configuration mode.
Step 14	**channel-group** *port-channel number* **Example:** `Router(config-if) channel-group 1`	Configures an EtherChannel interface to an EtherChannel group.

Note The EtherChannel-to-EtherChannel over MPLS (Bridged) Interworking mode is also supported under VLAN.

EtherChannel-to-EtherChannel over MPLS (Routed) Interworking

Configure L2VPN interworking on the upstream interfaces of the PE routers. For more information about configuring L2VPN interworking on the PE routers, see L2VPN Interworking Modes, page 16-2.

After configuring MPLS Forwarding, perform the following steps on the downstream interfaces of the PE routers:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **mpls label protocol ldp**
4. **interface loopback** *loopback-interface-number*
5. **ip address** *ip-address ip-subnet-mask*
6. **exit**
7. **pseudowire-class** *pw-class-name*
8. **encapsulation mpls**
9. **interworking ip**
10. **exit**
11. **interface port-channel** *number*
12. **xconnect** *peer-ip-address vc-id* **encapsulation mpls pw-class** *pw-class-name*
13. **interface GigabitEthernet** *slot* | *subslot* | *port*
14. **channel-group** *port-channel number*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable**	Changes the privilege level for the corresponding CLI session.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**mpls label protocol ldp** **Example:** `Router# mpls label protocol ldp`	Specifies that LDP is the default label distribution protocol.
Step 4	**interface loopback** *loopback-interface-number* **Example:** `Router# interface loopback 1`	Specifies the loopback interface, and enters interface configuration mode.
Step 5	**ip address** *ip-address mask* **Example:** `Router# ip address 10.10.2.1 255.255.255.0`	Sets the IP address and mask for the loopback interface.
Step 6	**exit**	Exits interface configuration mode.
Step 7	**pseudowire-class** *pw-class-name* **Example:** `Router(config)# pseudowire-class gec-bridged`	Specifies the name of a Layer 2 pseudowire class and enters pseudowire class configuration mode.
Step 8	**encapsulation mpls** **Example:** `Router(config-pw)# encapsulation mpls`	Uses MPLS as the tunneling method to encapsulate data in the pseudowire.
Step 9	**interworking ip** **Example:** `Router(config-pw)# interworking ip`	Enables the L2VPN Interworking feature, and causes IP packets to be extracted from the attachment circuit and sent over the pseudowire. Attachment circuit frames that do not contain IPv4 packets are dropped.
Step 10	**exit**	Exits xconnect configuration mode.
Step 11	**interface port-channel** *number* **Example:** `Router(config)# interface port-channel 1`	Creates an EtherChannel interface on the Cisco Cable Modem Termination System (CMTS).
Step 12	**xconnect** *peer-ip-address vc-id* **encapsulation mpls pseudowire-class** *pw-class-name* **Example:** `Router(config-if)# xconnect 10.0.0.1 707 encapsulation mpl pseudowire-class gec-routed`	Binds an attachment circuit to a pseudowire to configure an AToM static pseudowire, specifies MPLS as the tunneling method, and enters xconnect configuration mode.

	Command or Action	Purpose
Step 13	**interface GigabitEthernet** *slot* / *subslot* / *port* **Example:** `Router(config)# interface GigabitEthernet 0/0/1`	Specifies the Gigabit Ethernet interface, and enters interface configuration mode.
Step 14	**channel-group** *port-channel number* **Example:** `Router(config-if) channel-group 1`	Configures EtherChannel interfaces to an EtherChannel group.

Note The EtherChannel-to-EtherChannel over MPLS (Routed) Interworking mode is also supported under VLAN.

Example: GEC Like-to-Like (Routed) Interworking

The following example shows the configuration of the GEC Like-to-Like (Routed) Interworking feature:

```
no ip domain lookup
mpls label range 101 4000 static 4001 5001
mpls label protocol ldp
mpls ip default-route
mpls ldp graceful-restart
xconnect logging pseudowire status
!
pseudowire-class gec-bridged
encapsulation mpls
interworking ethernet!
pseudowire-class gec-routed
encapsulation mpls
interworking ip
!
interface Loopback0
ip address 44.1.1.2 255.255.255.255
!
interface GigabitEthernet0/0/1
ip address 10.10.1.2 255.255.255.0
negotiation auto
mpls ip
!
interface port-channel 1
xconnect 190.1.1.1 100 encapsulation mpls pw-class gec-bridged
!
interface GigabitEthernet0/0/3
channel-group 1
!
interface GigabitEthernet0/0/2
channel-group 1
!
router ospf 10
log-adjacency-changes
network 44.1.1.2 0.0.0.0 area 0
network 10.10.1.2 0.0.0.255 area 0
```

Any-to-EtherChannel over MPLS (Bridged) Interworking

You can configure Any-to-EtherChannel over MPLS (Bridged) interworking on the Cisco ASR 1000 Series Routers.

Any-to-EtherChannel over MPLS (Bridged) interworking supports the following modes:

- Frame Relay-to-EtherChannel
- ATM-to-EtherChannel
- Ethernet-to-EtherChannel

Irrespective of the mode used, in Any-to-EtherChannel over MPLS (Bridged) interworking, configure L2VPN interworking on the upstream interfaces of PE routers. For more information about configuring L2VPN interworking on the PE routers, see L2VPN Interworking Modes, page 16-2.

For information about how to configure Frame Relay or ATM on the downstream interfaces of a PE router, see the Configuring Frame Relay DLCI-to-ATM AAL5SNAP Bridged Interworking, page 16-15. Meanwhile, perform the steps described in the EtherChannel-to-EtherChannel over MPLS (Bridged) Interworking, page 16-28, on the downstream interfaces of the other PE router.

For the Ethernet-to-EtherChannel mode, perform the steps described in EtherChannel-to-EtherChannel over MPLS (Bridged) Interworking, page 16-28 on the downstream interfaces of the other PE router. Meanwhile, perform the following steps on the downstream interfaces of the PE routers:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **mpls label protocol ldp**
4. **interface loopback** *loopback-interface-number*
5. **ip address** *ip-address ip-subnet-mask*
6. **exit**
7. **pseudowire-class** *pw-class-name*
8. **encapsulation mpls**
9. **interworking ethernet**
10. **interface GigabitEthernet** *slot* | *subslot* | *port*
11. **xconnect** *peer-ip-address vc-id* **encapsulation mpls** *pw-class pw-class-name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable**	Changes the privilege level for the corresponding CLI session.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**mpls label protocol ldp** **Example:** `Router# mpls label protocol ldp`	Specifies that LDP is the default label distribution protocol.
Step 4	**interface loopback** *loopback-interface-number* **Example:** `Router# interface loopback 1`	Specifies the loopback interface, and enters the interface configuration mode.
Step 5	**ip address** *ip-address mask* **Example:** `Router# ip address 10.10.2.1 255.255.255.0`	Sets the IP address and mask for the loopback interface.
Step 6	**exit**	Exits interface configuration mode.
Step 7	**pseudowire-class** *pw-class-name* **Example:** `Router(config)# pseudowire-class gec-bridged`	Specifies the name of a Layer 2 pseudowire class and enters the pseudowire class configuration mode.
Step 8	**encapsulation mpls** **Example:** `Router(config-pw)# encapsulation mpls`	Uses MPLS as the tunneling method to encapsulate data in the pseudowire.
Step 9	**interworking ethernet** **Example:** `Router(config-pw)# interworking ethernet`	Enables the L2VPN Interworking feature, and causes Ethernet frames to be extracted from the attachment circuit and sent over the pseudowire. Ethernet end-to-end transmission is assumed. Attachment circuit frames that do not contain Ethernet frames are dropped. In the case of VLAN, the VLAN tag is removed, which leaves a pure Ethernet frame.
Step 10	**interface GigabitEthernet** *slot* / *subslot* / *port* **Example:** `Router(config)# interface GigabitEthernet 0/0/1`	Specifies the Gigabit Ethernet interface, and enters interface configuration mode.
Step 11	**xconnect** *peer-ip-address vc-id* **encapsulation mpls pseudowire-class** *pw-class-name* **Example:** `Router(config-if)# xconnect 10.0.0.1 707 encapsulation mpl pseudowire-class gec-bridged`	Binds an attachment circuit to a pseudowire to configure an AToM static pseudowire, specifies MPLS as the tunneling method, and enters the xconnect configuration mode.

Note Ethernet-to-EtherChannel over MPLS (Bridge) Interworking mode is also supported under VLAN.

Any-to-EtherChannel over MPLS (Routed) Interworking

You can configure Any-to-EtherChannel over MPLS (Routed) interworking on the Cisco ASR 1000 Series Routers.

Any-to-EtherChannel over MPLS (Routed) interworking supports the following modes:

- ATM-to-EtherChannel
- Ethernet-to-EtherChannel
- PPP-to-EtherChannel

Configure L2VPN interworking on the upstream interfaces of PE routers. For more information about configuring L2VPN interworking on the PE routers, see L2VPN Interworking Modes, page 16-2.

For information about how to configure ATM on the downstream interfaces of a PE router, see the Configuring Frame Relay DLCI-to-ATM AAL5SNAP Bridged Interworking, page 16-15. Meanwhile, perform the steps described in the EtherChannel-to-EtherChannel over MPLS (Routed) Interworking, page 16-30, on the downstream interfaces of the other PE router.

For the Ethernet-to-EtherChannel mode, see the Any-to-EtherChannel over MPLS (Bridged) Interworking, page 16-33.

For the PPP-to-EtherChannel mode, perform the steps described in the EtherChannel-to-EtherChannel over MPLS (Routed) Interworking, page 16-30, on the downstream interfaces of the other PE router. Meanwhile, perform the following steps on the downstream interfaces of the PE routers:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. (Optional) **ipv6 unicast-routing**
4. **mpls ip default-route**
5. **mpls ldp graceful-restart**
6. **xconnect logging pseudowire status**
7. **controller t1** *slot* | *subslot* | *port*
8. **clock source internal**
9. **linecode b8zs**
10. **cablelength** *long db-loss-value*
11. **channel-group** *channel-group-number timeslots range*
12. **exit**
13. **pseudowire-class** *pw-class-name*
14. **encapsulation mpls**
15. **interworking ethernet**
16. **exit**

17. **interface loopback** *loopback-interface-number*
18. **ip address** *ip-address mask*
19. **exit**
20. **interface serial** *slot | subslot | port:timeslot*
21. **no ip address**
22. **encapsulation ppp**
23. **clock source internal**
24. **xconnect** *vc-id* **pw-class** *pw-class pw-class-name*
25. **xconnect** *peer-loopback vc-id pw-class pe-class-name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. Enter your password, if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**ipv6 unicast-routing** **Example:** `Router# ipv6 unicast-routing`	(Optional) Enables the task of forwarding the IPv6 unicast datagrams.
Step 4	**mpls ip default-route** **Example:** `Router(config)# mpls ip default-route`	Enables the distribution of labels associated with the IP default route.
Step 5	**mpls ldp graceful-restart** **Example:** `Router(config)# mpls ldp graceful-restart`	Enables MPLS LDP graceful restart.
Step 6	**xconnect logging pseudowire status** **Example:** `Router(config)# xconnect logging pseudowire status`	Enables system logging (syslog) reporting of pseudowire status events.
Step 7	**controller t1 slot/subslot/port** **Example:** `Router(config)# controller T1 0/3/0`	Configures a T1 controller and enters controller configuration mode.
Step 8	**clock source internal** **Example:** `Router(config-controller)# clock source internal`	Configures the clock source of a DS1 link and uses the internal clock from the interface.
Step 9	**linecode b8zs** **Example:** `Router(config-controller)# linecode b8zs`	Specifies Binary 8-Zero Substitution (B8ZS) as the line code type for the T1 controller.
Step 10	**cablelength long db-loss-value** **Example:** `Router(config-controller)# cablelength long 0db`	Decreases the transmit signal by 0 dB. This is the default value.

	Command or Action	Purpose
Step 11	**channel-group** ***channel-group-number*** **timeslots** ***range*** **Example:** **Router(config-controller)# channel-group 0 timeslots 1-24**	Configures serial WAN on a T1 or E1 interface.
Step 12	**exit**	Exits pseudowire class configuration mode.
Step 13	**pseudowire-class** [*pw-class-name*] **Example:** Router(config)# **pseudowire-class** atm-fr-bridged	Establishes a pseudowire class with a name that you specify and enters the pseudowire class configuration mode.
Step 14	**encapsulation mpls** **Example:** Router(config-pw-class)# **encapsulation mpls**	Enables MPLS encapsulation on the interface.
Step 15	**interworking ethernet** **Example:** Router(config-pw-class)# **interworking ethernet**	Enables the L2VPN Ethernet interworking feature.
Step 16	**exit**	Exits pseudowire class configuration mode.
Step 17	**interface loopback** *loopback-interface-number* **Example:** Router(config)# **interface loopback** 0	Specifies the Loopback logical interface.
Step 18	**ip address** *ip-address mask* **Example:** Router(config-if)# **ip address** 44.1.1.2 255.255.255.255	Specifies the IP address for the Loopback interface.
Step 19	**exit**	Exits the interface configuration mode.
Step 20	**interface serial** slot/subslot/port:timeslot **Example:** Router(config)# **interface Serial**0/3/0:0	Specifies a serial interface created on a channelized T1 controller.
Step 21	**no ip address** **Example:** Router(config-if)# **no ip address**	Removes the previously configured IP address.
Step 22	**encapsulation ppp** **Example:** Router(config-if)# **encapsulation frame-relay**	Configures the PPP (for serial interface) encapsulation on serial interface.

	Command or Action	Purpose
Step 23	`clock source internal`	Specifies that the T1/E1 link uses the internal clock from the interface.
Step 24	`xconnect` *peer-loopback vc-id pw-class pe-class-name*	Binds an attachment circuit to a pseudowire to configure an AToM static pseudowire, specifies MPLS as the tunneling method, and enters xconnect configuration mode.

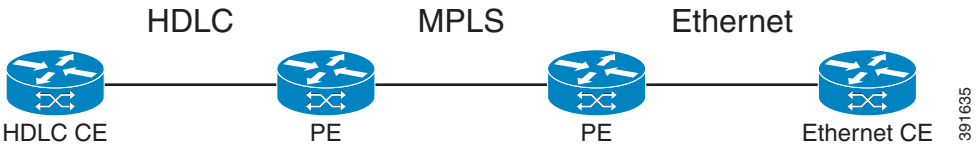
Note Ethernet-to-EtherChannel over MPLS (Bridge) Interworking mode is also supported under VLAN.

High-Level Data Link Control-Ethernet Interworking

HDLC-Ethernet over MPLS is part of Any Transport over MPLS (AToM) solution. High-Level Link Control (HDLC) and Ethernet are two link-layer transports that utilize the AToM architecture. This section describes how these two transport types can communicate with each other using the AToM framework.

Figure 16-7 shows the topology of the HDLC-Ethernet Interworking feature.

Figure 16-7 Topology of the HDLC-Ethernet Interworking Feature



The following features are supported from Cisco IOS XE Release 3.13.0S on the Cisco ASR 1000 Series Aggregation Services Routers:

- HDLC-Ethernet Bridged-Mode Interworking
- HDLC-Ethernet Routed-Mode Interworking
- HDLC Encapsulation: CISCO
- Ethernet Encapsulation: Dot1Q, QinQ, Port Interface

Prerequisites for HDLC-Ethernet Interworking

Perform the following tasks to enable HDLC-Ethernet interworking:

- Configure a controller slot on Ethernet CE:

```
controller E1 2/0
channel-group 0 timeslots 1
no shutdown
interface Serial2/0:0
no shutdown
```

- Configure an Ethernet CE interface for Ethernet interworking:

```
bridge irb
```

```
bridge 1 protocol ieee
bridge 1 route ip

interface Serial2/0:0
no bridge-group 1
no ip address
!
int BVI1
no ip address
ip address 192.168.1.1 255.255.255.0
no shut
!
interface Serial2/0:0
description Connect to PE1
no ip address
encapsulation hdlc
bridge-group 1
no shut
```

- Configure an Ethernet CE interface for IP interworking:

```
interface Serial2/0:0
description Connect to PE1
ip address 192.168.1.1 255.255.255.0
encapsulation hdlc
no shut
```

Restrictions for HDLC-Ethernet Interworking

The following features are not supported:

- HDLC encapsulation: none CISCO
- IPv6 is not supported in routed mode

Configuring HDLC-Ethernet Interworking

HDLC-Ethernet Interworking can be configured in the following two modes:

- Bridge Mode, page 16-40
- Routed Mode, page 16-46

Bridge Mode

Perform the following steps to configure the HDLC-Ethernet Interworking in the bridge mode via interface-based configuration:

On the HDLC-PE

SUMMARY STEPS

1. **enable**
2. **configure terminal**

3. **pseudowire-class** [*pw-class-name*]
4. **encapsulation mpls**
5. **interworking ethernet**
6. **interface serial** *slot* | *subslot* | *port*
7. **no ip address**
8. **xconnect** *peer-ip-address vc-id* **pseudowire-class** *pw-class-name*
9. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable**	Changes the privilege level for the corresponding CLI session.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**pseudowire-class** *pw-class-name* **Example:** `Router(config)# pseudowire-class pw-iw-ether`	Specifies the name of a Layer 2 pseudowire class and enters pseudowire class configuration mode.
Step 4	**encapsulation mpls** **Example:** `Router(config-pw)# encapsulation mpls`	Uses MPLS as the tunneling method to encapsulate data in the pseudowire.
Step 5	**interworking ethernet** **Example:** `Router(config-pw)# interworking ethernet`	Enables the L2VPN Interworking feature, and causes Ethernet frames to be extracted from the attachment circuit and sent over the pseudowire. Ethernet end-to-end transmission is assumed. Attachment circuit frames that do not contain Ethernet frames are dropped. In the case of VLAN, the VLAN tag is removed, which leaves a pure Ethernet frame.
Step 6	**interface serial** *slot* / *subslot* / *port* **Example:** `Router(config)# interface Serial0/1/0:0`	Specifies the serial interface, and enters interface configuration mode.
Step 7	**no ip address** **Example:** `Router(config-if)# no ip address`	Removes all the IP addresses of the interface.
Step 8	**xconnect** *peer-ip-address vc-id* **pseudowire-class** *pw-class-name* **Example:** `Router(config-if)# xconnect 17.17.17.17 100 pw-class pw-iw-ether`	Binds an attachment circuit to a pseudowire to configure an AToM static pseudowire, specifies MPLS as the tunneling method, and enters xconnect configuration mode.

On the Ethernet PE

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **pseudowire-class** [*pw-class-name*]

4. **encapsulation mpls**
5. **interworking ethernet**
6. **interface GigabitEthernet** *slot | subslot | port*
7. **encapsulation dot1Q** *vlan-id*
8. **xconnect** *peer-ip-address vc-id* **pseudowire-class** *pw-class-name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable**	Changes the privilege level for the corresponding CLI session.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**pseudowire-class** *pw-class-name* **Example:** `Router(config)# pseudowire-class pw-iw-ether`	Specifies the name of a Layer 2 pseudowire class and enters pseudowire class configuration mode.
Step 4	**encapsulation mpls** **Example:** `Router(config-pw)# encapsulation mpls`	Uses MPLS as the tunneling method to encapsulate data in the pseudowire.
Step 5	**interworking ethernet** **Example:** `Router(config-pw)# interworking ethernet`	Enables the L2VPN Interworking feature, and causes Ethernet frames to be extracted from the attachment circuit and sent over the pseudowire. Ethernet end-to-end transmission is assumed. Attachment circuit frames that do not contain Ethernet frames are dropped. In the case of VLAN, the VLAN tag is removed, which leaves a pure Ethernet frame.
Step 6	**interface GigabitEthernet** *slot	subslot	port* **Example:** `Router(config)# interface GigabitEthernet0/0/0.3`	Specifies the Gigabit Ethernet interface, and enters interface configuration mode.
Step 7	**encapsulation dot1Q** *vlan-id* **Example:** `Router(config-if)# encapsulation dot1Q 3`	Removes all the IP addresses of the interface.
Step 8	**xconnect** *peer-ip-address vc-id* **pseudowire-class** *pw-class-name* **Example:** `Router(config-if)# xconnect 16.16.16.16 100 pseudowire-class pw-iw-ether`	Binds an attachment circuit to a pseudowire to configure an AToM static pseudowire, specifies MPLS as the tunneling method, and enters xconnect configuration mode.

Perform the following steps to configure the HDLC-Ethernet Interworking in the bridge mode via protocol-based configuration:

On the HDLC-PE

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **l2vpn xconnect context** *xc-name*
4. **interworking ethernet**
5. **member** *interface-id*
6. **member** *ip-address vc-id* **encapsulation mpls**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable**	Changes the privilege level for the corresponding CLI session.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**l2vpn xconnect context** *xc-name* **Example:** `Router(config)# l2vpn xconnect context HDLC`	Creates an empty cross-connect, and enters xconnect submode.
Step 4	**interworking ethernet** **Example:** `Router(config)# interworking ethernet`	Enables the L2VPN Interworking feature, and causes Ethernet frames to be extracted from the attachment circuit and sent over the pseudowire. Ethernet end-to-end transmission is assumed. Attachment circuit frames that do not contain Ethernet frames are dropped. In the case of VLAN, the VLAN tag is removed, which leaves a pure Ethernet frame.
Step 5	**member** *interface-id* **Example:** `Router(config)# member serial0/1/0:0`	Adds an interface as an AC segment of xconnect.
Step 6	**member** *ip-address vc-id* **encapsulation mpls** **Example:** `Router(config)# member 17.17.17.17 100 encapsulation mpls`	Adds a pseudowire member to xconnect.

On the Ethernet PE

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **l2vpn xconnect context foo**
4. **interworking ethernet**
5. **member** *interface-id*
6. **member** *ip-address vc-id* **encapsulation mpls**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable**	Changes the privilege level for the corresponding CLI session.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**l2vpn xconnect context foo** **Example:** `Router(config)# l2vpn xconnect context foo`	Creates an empty cross-connect, and enters xconnect submode.
Step 4	**interworking ethernet** **Example:** `Router(config)# interworking ethernet`	Enables the L2VPN Interworking feature, and causes Ethernet frames to be extracted from the attachment circuit and sent over the pseudowire. Ethernet end-to-end transmission is assumed. Attachment circuit frames that do not contain Ethernet frames are dropped. In the case of VLAN, the VLAN tag is removed, which leaves a pure Ethernet frame.
Step 5	**member** *interface-id* **Example:** `Router(config)# member GigabitEthernet0/0/0.3`	Adds an interface as an AC segment of xconnect.
Step 6	**member** *ip-address vc-id* **encapsulation mpls** **Example:** `Router(config)# member 16.16.16.16 100 encapsulation mpls`	Adds a pseudowire member to xconnect.

Routed Mode

Perform the following steps to configure the HDLC-Ethernet Interworking in the routed mode via interface-based configuration:

On HDLC-PE

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **pseudowire-class** [*pw-class-name*]
4. **encapsulation mpls**
5. **interworking ip**
6. **interface serial** *slot* | *subslot* | *port*

7. **no ip address**
8. **xconnect** *peer-ip-address vc-id* **pseudowire-class** *pw-class-name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable**	Changes the privilege level for the corresponding CLI session.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**pseudowire-class** *pw-class-name* **Example:** `Router(config)# pseudowire-class pw-ip-ether`	Specifies the name of a Layer 2 pseudowire class and enters pseudowire class configuration mode.
Step 4	**encapsulation mpls** **Example:** `Router(config-pw)# encapsulation mpls`	Uses MPLS as the tunneling method to encapsulate data in the pseudowire.
Step 5	**interworking ip** **Example:** `Router(config-pw)# interworking ip`	Enables the L2VPN Interworking feature, and causes IP packets to be extracted from the attachment circuit and sent over the pseudowire. Attachment circuit frames that do not contain IPv4 packets are dropped.
Step 6	**interface serial** *slot* / *subslot* / *port* **Example:** `Router(config)# interface Serial0/1/0:0`	Specifies the serial interface, and enters interface configuration mode.
Step 7	**no ip address** **Example:** `Router(config-if)# no ip address`	Removes all the IP addresses of the interface.
Step 8	**xconnect** *peer-ip-address vc-id* **pseudowire-class** *pw-class-name* **Example:** `Router(config-if)# xconnect 17.17.17.17 100 pw-class pw-ip-ether`	Binds an attachment circuit to a pseudowire to configure an AToM static pseudowire, specifies MPLS as the tunneling method, and enters xconnect configuration mode.

On Ethernet PE

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **pseudowire-class** [*pw-class-name*]

4. **encapsulation mpls**
5. **interworking ip**
6. **interface GigabitEthernet** *slot | subslot | port*
7. **encapsulation dot1Q** *vlan-id*
8. **xconnect** *peer-ip-address vc-id* **pseudowire-class** *pw-class-name*
9. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable**	Changes the privilege level for the corresponding CLI session.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**pseudowire-class** *pw-class-name* **Example:** `Router(config)# pseudowire-class pw-ip-ether`	Specifies the name of a Layer 2 pseudowire class and enters pseudowire class configuration mode.
Step 4	**encapsulation mpls** **Example:** `Router(config-pw)# encapsulation mpls`	Uses MPLS as the tunneling method to encapsulate data in the pseudowire.
Step 5	**interworking ip** **Example:** `Router(config-pw)# interworking ip`	Enables the L2VPN Interworking feature, and causes IP packets to be extracted from the attachment circuit and sent over the pseudowire. Attachment circuit frames that do not contain IPv4 packets are dropped.
Step 6	**interface GigabitEthernet** *slot	subslot	port* **Example:** `Router(config)# interface GigabitEthernet0/0/0.3`	Specifies the Gigabit Ethernet interface, and enters interface configuration mode.
Step 7	**encapsulation dot1Q** *vlan-id* **Example:** `Router(config-if)# encapsulation dot1Q 3`	Enables IEEE 802.1Q encapsulation of traffic on a specified subinterface in a VLAN.
Step 8	**xconnect** *peer-ip-address vc-id* **pseudowire-class** *pw-class-name* **Example:** `Router(config-if)# xconnect 16.16.16.16 100 pseudowire-class pw-ip-ether`	Binds an attachment circuit to a pseudowire to configure an AToM static pseudowire, specifies MPLS as the tunneling method, and enters xconnect configuration mode.

Perform the following steps to configure the HDLC-Ethernet Interworking in the routed mode via protocol-based configuration:

On HDLC-PE

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **l2vpn xconnect context foo**
4. **interworking ip**
5. **member** *interface-id*
6. **member** *ip-address vc-id* **encapsulation mpls**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable**	Changes the privilege level for the corresponding CLI session.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**l2vpn xconnect context foo** **Example:** `Router(config)# l2vpn xconnect context foo`	Creates an empty cross-connect, and enters xconnect submode.
Step 4	**interworking ip** **Example:** `Router(config)# interworking ip`	Enables the L2VPN Interworking feature, and causes IP packets to be extracted from the attachment circuit and sent over the pseudowire. Attachment circuit frames that do not contain IPv4 packets are dropped.
Step 5	**member** *interface-id* **Example:** `Router(config)# member serial0/1/0:0`	Adds an interface as an AC segment of xconnect.
Step 6	**member** *ip-address* **encapsulation mpls** **Example:** `Router(config)# member 17.17.17.17 100 encapsulation mpls`	Adds a pseudowire member to xconnect.

On Ethernet PE

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **l2vpn xconnect context foo**
4. **interworking ip**
5. **member** *interface-id*
6. **member** *ip-address vc-id* **encapsulation mpls**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable**	Changes the privilege level for the corresponding CLI session.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**l2vpn xconnect context foo** **Example:** `Router(config)# l2vpn xconnect context foo`	Creates an empty cross-connect, and enters xconnect submode.
Step 4	**interworking ip** **Example:** `Router(config)# interworking ip`	Enables the L2VPN Interworking feature, and causes IP packets to be extracted from the attachment circuit and sent over the pseudowire. Attachment circuit frames that do not contain IPv4 packets are dropped.
Step 5	**member** *interface-id* **Example:** `Router(config)# member GigabitEthernet0/0/0.3`	Adds an interface as an AC segment of xconnect.
Step 6	**member** *ip-address vcid* **encapsulation mpls** **Example:** `Router(config)# member 16.16.16.16 100 encapsulation mpls`	Adds a pseudowire member to xconnect.

Example: HDLC-Ethernet Interworking Configuration

The following examples show how to configure the HDLC-Ethernet Interworking feature, and verify the configuration using **show** commands for legacy and new protocol-based outputs on the Cisco ASR 1000 Series Aggregation Services Routers:

- Example: Different Forms of Protocol-Based CLI Configuration, page 16-51
- Example: Verifying the Configuration for HDLC-Ethernet Interworking, page 16-57
- Example: HDLC-Dot1Q Interworking, page 16-67

Example: Different Forms of Protocol-Based CLI Configuration

The following example shows how to configure the HDLC-Ethernet interworking on the Controller slot on Ethernet CE:

```
controller E1 2/0
channel-group 0 timeslots 1
no shutdown
interface Serial2/0:0
no shutdown
```

The following example shows how to configure the HDLC-Ethernet interworking on the Controller slot on Ethernet PE:

```
controller E1 0/1/0
channel-group 0 timeslots 1
no shutdown
interface Serial0/1/0:0
no shutdown
```

The following example shows how to configure the HDLC-Ethernet interworking using legacy CLI.

The following example shows how to configure on HDLC-CE and HDLC-PE in Bridged (Ethernet) mode using legacy CLI:

On HDLC-CE

```
configure terminal
bridge irb
bridge 1 protocol ieee
bridge 1 route ip
!
int BVI1
ip address 192.168.1.1 255.255.255.0
no shut
!
interface Serial2/0:0
description Connect to PE1
encapsulation hdlc
bridge-group 1
no shut
end
```

HDLC-PE:

```
configure terminal
pseudowire-class pw-iw-eth
 encapsulation mpls
 interworking Ethernet
!
interface Serial0/1/0:0
 description Connect to CE1
 encapsulation hdlc
 no ip address
 xconnect 3.3.3.3 100 pw-class pw-iw-eth
 no shut
end
```

The following example shows how to configure Ethernet on Ethernet-CE and Ethernet-PE in Bridged (Ethernet) mode using legacy CLI:

On Ethernet-CE

```
configure terminal
interface GigabitEthernet0/1
description Connect to PE2
ip address 192.168.1.2 255.255.255.0
ip irdp
ip irdp maxadvertinterval 4
no shut
end
```

On Ethernet-PE

```
configure terminal
pseudowire-class pw-iw-eth
```

```
 encapsulation mpls
 interworking Ethernet
!
interface GigabitEthernet1/0/0
 description Connect to CE2
 no ip address
xconnect 1.1.1.1 100 pw-class pw-iw-eth
 no shut
end
```

The following example shows how to configure VLAN at Ethernet-CE and Ethernet-PE in Bridged (Ethernet) mode using legacy CLI:

On Ethernet-CE:

```
configure terminal
interface GigabitEthernet0/1
no ip address
no shut
!
interface GigabitEthernet0/1.10
description Connect to PE2
encapsulation dot1q 10
ip address 192.168.1.2 255.255.255.0
ip irdp
ip irdp maxadvertinterval 4
no shut
end
```

On Ethernet-PE:

```
configure terminal
pseudowire-class pw-iw-eth
 encapsulation mpls
 interworking Ethernet
!
interface GigabitEthernet1/0/0
no ip address
no shut
!
interface GigabitEthernet1/0/0.10
 description Connect to CE2
 encapsulation dot1Q 10
 no ip address
 xconnect 1.1.1.1 100 pw-class pw-iw-eth
 no shut
end
```

The following example shows how to configureQinQ at Ethernet-CE and Ethernet-PE in Bridged (Ethernet) mode using legacy CLI:

On Ethernet-CE:

```
configure terminal
interface GigabitEthernet0/1
no ip address
no shut
!
interface GigabitEthernet0/1.10
 description Connect to PE2
encapsulation dot1q 10 second-dot1q 20
ip address 192.168.1.2 255.255.255.0
ip irdp
ip irdp maxadvertinterval 4
no shut
```

```
end
```

On Ethernet-PE:

```
configure terminal
pseudowire-class pw-iw-eth
 encapsulation mpls
 interworking Ethernet
!
interface GigabitEthernet1/0/0
no ip address
no shut
!
interface GigabitEthernet1/0/0.10
 description Connect to CE2
 encapsulation dot1Q 10 second-dot1q 20
 no ip address
 xconnect 1.1.1.1 100 pw-class pw-iw-eth
 no shut
end
```

The following example shows how to configure HDLC-Ethernet interworking using Protocol-Based CLI:

The following example shows how to configure HDLC on HDLC-CE and HDLC-PE in Bridged (Ethernet) mode using protocol-based CLI:

On HDLC-CE:

```
configure terminal
bridge irb
bridge 1 protocol ieee
bridge 1 route ip
!
int BVI1
ip address 192.168.1.1 255.255.255.0
no shut
!
interface Serial2/0:0
description Connect to PE1
encapsulation hdlc
bridge-group 1
no shut
end
```

On HDLC-PE:

```
configure terminal
interface Serial0/1/0:0
description Connect to CE1
encapsulation hdlc
no ip addres
no shut
!
Interface pseudowire101
encapsulation mpls
neighbor 3.3.3.3 100
signaling protocol ldp
no shut

l2vpn xconnect context foohdlc
interworking ethernet
member Serial0/1/0:0
member pseudowire101
no shut
```

```
end
```

The following example shows how to configure Ethernet on Ethernet-CE and Ethernet-PE - Bridged (Ethernet) mode using protocol-based CLI:

On Ethernet-CE:

```
configure terminal
interface GigabitEthernet0/1
description Connect to PE2
ip address 192.168.1.2 255.255.255.0
ip irdp
ip irdp maxadvertinterval 4
no shut
end
```

On Ethernet-PE:

```
configure terminal
interface GigabitEthernet1/0/0
 description Connect to CE2
no ip address
no shut
!
Interface pseudowire101
encapsulation mpls
neighbor 1.1.1.1 100
signaling protocol ldp
no shut
!
l2vpn xconnect context fooeth
interworking ethernet
member GigabitEthernet1/0/0
member pseudowire101
no shut
end
```

The following example shows how to configure VLAN on Ether-CE and Ethernet-PE in Bridged (Ethernet) mode using protocol-based CLI:

On Ethernet-CE:

```
configure terminal
!
interface GigabitEthernet0/1
no ip address
no shut
!
interface GigabitEthernet0/1.10
encapsulation dot1q 10
description Connect to PE2
ip address 192.168.1.2 255.255.255.0
ip irdp
ip irdp maxadvertinterval 4
no shut
end
```

On Ethernet-PE:

```
configure terminal
!
interface GigabitEthernet1/0/0
no ip address
no shut
!
```

```
interface GigabitEthernet1/0/0.10
description Connect to CE2
encapsulation dot1q 10
no ip addres
no shut
!
Interface pseudowire101
encapsulation mpls
neighbor 1.1.1.1 100
signaling protocol ldp
no shut
!

l2vpn xconnect context foovlan
interworking ethernet
member GigabitEthernet1/0/0.10
member pseudowire101
no shut
end
```

The following example shows how to configure QinQ on Ethernet-CE and Ethernet-PE in Bridged (Ethernet) mode using protocol-based CLI:

Ethernet-CE:

```
configure terminal
!
interface GigabitEthernet0/1
no ip address
no shut
!
interface GigabitEthernet0/1.10
description Connect to PE2
encapsulation dot1q 10 second-dot1q 20
ip address 192.168.1.2 255.255.255.0
ip irdp
ip irdp maxadvertinterval 4
no shut
end
```

Ethernet-PE:

```
configure terminal
!
interface GigabitEthernet1/0/0
no ip address
no shut
!
interface GigabitEthernet1/0/0.10
description Connect to CE2
encapsulation dot1q 10 second-dot1q 20
no ip addres
no shut
!
Interface pseudowire101
encapsulation mpls
neighbor 1.1.1.1 100
signaling protocol ldp
no shut
!
l2vpn xconnect context fooqinq
interworking ethernet
member GigabitEthernet1/0/0.10
member pseudowire101
```

```
no shut
end
```

Example: Verifying the Configuration for HDLC-Ethernet Interworking

Use the following **show** commands to verify the configuration for HDLC-Ethernet interworking:

Port mode

The following example shows how to verify the HDLC configuration on PE:

```
Router# show mpls l2transport vc

Local intf     Local circuit              Dest address    VC ID      Status
-------------  -------------------------- --------------- ---------- ----------
Se0/1/0:0      HDLC                       104.0.0.1       101        UP

Router# show mpls l2transport vc detail

Local interface: Se0/1/0:0 up, line protocol up, HDLC up
  Interworking type is Ethernet
  Destination address: 104.0.0.1, VC ID: 101, VC status: up
    Output interface: Fa0/0/1, imposed label stack {20 22}
    Preferred path: not configured
    Default path: active
    Next hop: 10.1.1.2
  Create time: 00:00:19, last status change time: 00:00:15
    Last label FSM state change time: 00:00:15
  Signaling protocol: LDP, peer 104.0.0.1:0 up
    Targeted Hello: 102.0.0.1(LDP Id) -> 104.0.0.1, LDP is UP
    Graceful restart: configured and enabled
    Non stop routing: not configured and not enabled
    Status TLV support (local/remote)   : enabled/supported
      LDP route watch                   : enabled
      Label/status state machine        : established, LruRru
      Last local dataplane   status rcvd: No fault
      Last BFD dataplane     status rcvd: Not sent
      Last BFD peer monitor  status rcvd: No fault
      Last local AC  circuit status rcvd: No fault
      Last local AC  circuit status sent: No fault
      Last local PW i/f circ status rcvd: No fault
      Last local LDP TLV     status sent: No fault
      Last remote LDP TLV    status rcvd: No fault
      Last remote LDP ADJ    status rcvd: No fault
    MPLS VC labels: local 33, remote 22
    Group ID: local 0, remote 0
    MTU: local 1500, remote 1500
    Remote interface description: Connect to CE2
  Sequencing: receive disabled, send disabled
  Control Word: On
  SSO Descriptor: 104.0.0.1/101, local label: 33
  Dataplane:
    SSM segment/switch IDs: 4274/4273 (used), PWID: 26
  VC statistics:
    transit packet totals: receive 3, send 6
    transit byte totals:   receive 162, send 366
    transit packet drops:  receive 0, seq error 0, send 0

Router# show l2vpn atom vc
                                     Service
Interface Peer ID         VC ID      Type   Name                     Status
--------- --------------- ---------- ------ ------------------------ ----------
```

```
pw101     104.0.0.1       101        p2p    foo101                  UP

Router# show l2vpn atom vc detail

pseudowire101 is up, VC status is up PW type: Ethernet
  Create time: 00:00:18, last status change time: 00:00:14
    Last label FSM state change time: 00:00:14
  Destination address: 104.0.0.1 VC ID: 101
    Output interface: Fa0/0/1, imposed label stack {16 17}
    Preferred path: not configured
    Default path: active
    Next hop: 10.1.1.2
  Member of xconnect service foo101
    Associated member Se0/1/0:0 is up, status is up
    Interworking type is Ethernet
    Service id: 0xde000002
  Signaling protocol: LDP, peer 104.0.0.1:0 up
    Targeted Hello: 102.0.0.1(LDP Id) -> 104.0.0.1, LDP is UP
    Graceful restart: configured and enabled
    Non stop routing: not configured and not enabled
    PWid FEC (128), VC ID: 101
    Status TLV support (local/remote)         : enabled/supported
      LDP route watch                         : enabled
      Label/status state machine              : established, LruRru
      Local dataplane status received         : No fault
      BFD dataplane status received           : Not sent
      BFD peer monitor status received        : No fault
      Status received from access circuit     : No fault
      Status sent to access circuit           : No fault
      Status received from pseudowire i/f     : No fault
      Status sent to network peer             : No fault
      Status received from network peer       : No fault
      Adjacency status of remote peer         : No fault
  Sequencing: receive disabled, send disabled
  Bindings
    Parameter    Local                          Remote
    ------------ ------------------------------ ------------------------------
    Label        18                             17
    Group ID     0                              0
    Interface    Connect to CE1                 Connect to CE2
    MTU          1500                           1500
    Control word on (configured: autosense)     on
    PW type      Ethernet                       Ethernet
    VCCV CV type 0x02                           0x02
                   LSPV [2]                       LSPV [2]
    VCCV CC type 0x07                           0x07
                   CW [1], RA [2], TTL [3]        CW [1], RA [2], TTL [3]
    Status TLV   enabled                        supported
  SSO Descriptor: 104.0.0.1/101, local label: 18
  Dataplane:
    SSM segment/switch IDs: 4106/4105 (used), PWID: 2
  Rx Counters
    3 input transit packets, 162 bytes
    0 drops, 0 seq err
  Tx Counters
    5 output transit packets, 305 bytes
    0 drops
```

The following example shows how to verify the Ethernet configuration on PE:

```
Router# show mpls l2transport vc

Local intf     Local circuit              Dest address    VC ID      Status
-------------  -------------------------- --------------- ---------- ----------
```

```
Gi1/0/0        Ethernet                  102.0.0.1       101        UP

Router# show mpls l2transport vc detail

Local interface: Gi1/0/0 up, line protocol up, Ethernet up
  Destination address: 102.0.0.1, VC ID: 101, VC status: up
    Output interface: Fa0/0/1, imposed label stack {19 33}
    Preferred path: not configured
    Default path: active
    Next hop: 11.1.1.1
  Create time: 00:00:22, last status change time: 00:00:19
    Last label FSM state change time: 00:00:19
  Signaling protocol: LDP, peer 102.0.0.1:0 up
    Targeted Hello: 104.0.0.1(LDP Id) -> 102.0.0.1, LDP is UP
    Graceful restart: configured and enabled
    Non stop routing: not configured and not enabled
    Status TLV support (local/remote)   : enabled/supported
      LDP route watch                   : enabled
      Label/status state machine        : established, LruRru
      Last local dataplane   status rcvd: No fault
      Last BFD dataplane     status rcvd: Not sent
      Last BFD peer monitor  status rcvd: No fault
      Last local AC  circuit status rcvd: No fault
      Last local AC  circuit status sent: No fault
      Last local PW i/f circ status rcvd: No fault
      Last local LDP TLV     status sent: No fault
      Last remote LDP TLV    status rcvd: No fault
      Last remote LDP ADJ    status rcvd: No fault
    MPLS VC labels: local 22, remote 33
    Group ID: local 0, remote 0
    MTU: local 1500, remote 1500
    Remote interface description: Connect to CE1
  Sequencing: receive disabled, send disabled
  Control Word: On
  SSO Descriptor: 102.0.0.1/101, local label: 22
  Dataplane:
    SSM segment/switch IDs: 4574/4573 (used), PWID: 80
  VC statistics:
    transit packet totals: receive 9, send 5
    transit byte totals:   receive 315, send 380
    transit packet drops:  receive 0, seq error 0, send 0

Router# show l2vpn atom vc
                                     Service
Interface Peer ID         VC ID      Type   Name                      Status
--------- --------------- ---------- ------ ------------------------- ----------
pw101     102.0.0.1       101        p2p    foo101                    UP

Router# show l2vpn atom vc detail

pseudowire101 is up, VC status is up PW type: Ethernet
  Create time: 00:00:23, last status change time: 00:00:20
    Last label FSM state change time: 00:00:20
  Destination address: 102.0.0.1 VC ID: 101
    Output interface: Fa0/0/1, imposed label stack {18 18}
    Preferred path: not configured
    Default path: active
    Next hop: 11.1.1.1
  Member of xconnect service foo101
    Associated member Gi1/0/0 is up, status is up
    Interworking type is Like2Like
    Service id: 0xb5000004
  Signaling protocol: LDP, peer 102.0.0.1:0 up
    Targeted Hello: 104.0.0.1(LDP Id) -> 102.0.0.1, LDP is UP
```

```
    Graceful restart: configured and enabled
    Non stop routing: not configured and not enabled
    PWid FEC (128), VC ID: 101
    Status TLV support (local/remote)         : enabled/supported
      LDP route watch                         : enabled
      Label/status state machine              : established, LruRru
      Local dataplane status received         : No fault
      BFD dataplane status received           : Not sent
      BFD peer monitor status received        : No fault
      Status received from access circuit     : No fault
      Status sent to access circuit           : No fault
      Status received from pseudowire i/f     : No fault
      Status sent to network peer             : No fault
      Status received from network peer       : No fault
      Adjacency status of remote peer         : No fault
  Sequencing: receive disabled, send disabled
  Bindings
    Parameter    Local                          Remote
    ------------ ------------------------------ ------------------------------
    Label        17                             18
    Group ID     0                              0
    Interface    Connect to CE2                 Connect to CE1
    MTU          1500                           1500
    Control word on (configured: autosense)     on
    PW type      Ethernet                       Ethernet
    VCCV CV type 0x02                           0x02
                   LSPV [2]                       LSPV [2]
    VCCV CC type 0x07                           0x07
                   CW [1], RA [2], TTL [3]        CW [1], RA [2], TTL [3]
    Status TLV   enabled                        supported
  SSO Descriptor: 102.0.0.1/101, local label: 17
  Dataplane:
    SSM segment/switch IDs: 4126/4125 (used), PWID: 4
  Rx Counters
    5 input transit packets, 175 bytes
    0 drops, 0 seq err
  Tx Counters
    3 output transit packets, 228 bytes
    0 drops
```

VLAN (dot1q) mode

The following example shows how to verify the HDLC configuration on PE:

```
Router# show mpls l2transport vc

Local intf     Local circuit              Dest address    VC ID      Status
-------------  -------------------------- --------------- ---------- ----------
Se0/1/0:0      HDLC                       104.0.0.1       138        UP

Router# show mpls l2transport vc detail

Local interface: Se0/1/0:0 up, line protocol up, HDLC up
  Interworking type is Ethernet
  Destination address: 104.0.0.1, VC ID: 138, VC status: up
    Output interface: Fa0/0/1, imposed label stack {20 53}
    Preferred path: not configured
    Default path: active
    Next hop: 10.1.1.2
  Create time: 00:00:19, last status change time: 00:00:15
    Last label FSM state change time: 00:00:15
  Signaling protocol: LDP, peer 104.0.0.1:0 up
    Targeted Hello: 102.0.0.1(LDP Id) -> 104.0.0.1, LDP is UP
    Graceful restart: configured and enabled
```

```
    Non stop routing: not configured and not enabled
    Status TLV support (local/remote)   : enabled/supported
      LDP route watch                   : enabled
      Label/status state machine        : established, LruRru
      Last local dataplane   status rcvd: No fault
      Last BFD dataplane     status rcvd: Not sent
      Last BFD peer monitor  status rcvd: No fault
      Last local AC  circuit status rcvd: No fault
      Last local AC  circuit status sent: No fault
      Last local PW i/f circ status rcvd: No fault
      Last local LDP TLV     status sent: No fault
      Last remote LDP TLV    status rcvd: No fault
      Last remote LDP ADJ    status rcvd: No fault
    MPLS VC labels: local 35, remote 53
    Group ID: local 0, remote 0
    MTU: local 1500, remote 1500
    Remote interface description: Connect to CE2
  Sequencing: receive disabled, send disabled
  Control Word: On
  SSO Descriptor: 104.0.0.1/138, local label: 35
  Dataplane:
    SSM segment/switch IDs: 4486/4485 (used), PWID: 65
  VC statistics:
    transit packet totals: receive 4, send 3
    transit byte totals:   receive 1036, send 183
    transit packet drops:  receive 0, seq error 0, send 0

Router# show l2vpn atom vc
                                      Service
Interface Peer ID         VC ID      Type   Name                      Status
--------- --------------- ---------- ------ ------------------------- ----------
pw138     104.0.0.1       138        p2p    foo138                    UP

Router# show l2vpn atom vc detail

pseudowire138 is up, VC status is up PW type: Ethernet
  Create time: 00:00:18, last status change time: 00:00:14
    Last label FSM state change time: 00:00:14
  Destination address: 104.0.0.1 VC ID: 138
    Output interface: Fa0/0/1, imposed label stack {16 30}
    Preferred path: not configured
    Default path: active
    Next hop: 10.1.1.2
  Member of xconnect service foo138
    Associated member Se0/1/0:0 is up, status is up
    Interworking type is Ethernet
    Service id: 0x4000027
  Signaling protocol: LDP, peer 104.0.0.1:0 up
    Targeted Hello: 102.0.0.1(LDP Id) -> 104.0.0.1, LDP is UP
    Graceful restart: configured and enabled
    Non stop routing: not configured and not enabled
    PWid FEC (128), VC ID: 138
    Status TLV support (local/remote)         : enabled/supported
      LDP route watch                         : enabled
      Label/status state machine              : established, LruRru
      Local dataplane status received         : No fault
      BFD dataplane status received           : Not sent
      BFD peer monitor status received        : No fault
      Status received from access circuit     : No fault
      Status sent to access circuit           : No fault
      Status received from pseudowire i/f     : No fault
      Status sent to network peer             : No fault
      Status received from network peer       : No fault
      Adjacency status of remote peer         : No fault
```

```
  Sequencing: receive disabled, send disabled
  Bindings
    Parameter    Local                          Remote
    ------------ ------------------------------ ------------------------------
    Label        20                             30
    Group ID     0                              0
    Interface    Connect to CE1                 Connect to CE2
    MTU          1500                           1500
    Control word on (configured: autosense)     on
    PW type      Ethernet                       Ethernet
    VCCV CV type 0x02                           0x02
                   LSPV [2]                       LSPV [2]
    VCCV CC type 0x07                           0x07
                   CW [1], RA [2], TTL [3]        CW [1], RA [2], TTL [3]
    Status TLV   enabled                        supported
  SSO Descriptor: 104.0.0.1/138, local label: 20
  Dataplane:
    SSM segment/switch IDs: 4313/4312 (used), PWID: 41
  Rx Counters
    2 input transit packets, 108 bytes
    0 drops, 0 seq err
  Tx Counters
    3 output transit packets, 183 bytes
    0 drops
```

The following example shows how to verify the VLAN configuration on PE:

```
Router# show mpls l2transport vc

Local intf     Local circuit              Dest address    VC ID      Status
-------------  -------------------------- --------------- ---------- ----------
Gi1/0/0.10     Eth VLAN 10                102.0.0.1       138        UP

Router# show mpls l2transport vc detail

Local interface: Gi1/0/0.10 up, line protocol up, Eth VLAN 10 up
  Interworking type is Ethernet
  Destination address: 102.0.0.1, VC ID: 138, VC status: up
    Output interface: Fa0/0/1, imposed label stack {19 35}
    Preferred path: not configured
    Default path: active
    Next hop: 11.1.1.1
  Create time: 00:00:22, last status change time: 00:00:20
    Last label FSM state change time: 00:00:20
  Signaling protocol: LDP, peer 102.0.0.1:0 up
    Targeted Hello: 104.0.0.1(LDP Id) -> 102.0.0.1, LDP is UP
    Graceful restart: configured and enabled
    Non stop routing: not configured and not enabled
    Status TLV support (local/remote)   : enabled/supported
      LDP route watch                   : enabled
      Label/status state machine        : established, LruRru
      Last local dataplane   status rcvd: No fault
      Last BFD dataplane     status rcvd: Not sent
      Last BFD peer monitor  status rcvd: No fault
      Last local AC  circuit status rcvd: No fault
      Last local AC  circuit status sent: No fault
      Last local PW i/f circ status rcvd: No fault
      Last local LDP TLV     status sent: No fault
      Last remote LDP TLV    status rcvd: No fault
      Last remote LDP ADJ    status rcvd: No fault
    MPLS VC labels: local 53, remote 35
    Group ID: local 0, remote 0
    MTU: local 1500, remote 1500
```

```
    Remote interface description: Connect to CE1
  Sequencing: receive disabled, send disabled
  Control Word: On
  SSO Descriptor: 102.0.0.1/138, local label: 53
  Dataplane:
    SSM segment/switch IDs: 4784/4783 (used), PWID: 117
  VC statistics:
    transit packet totals: receive 6, send 6
    transit byte totals:   receive 234, send 1276
    transit packet drops:  receive 0, seq error 0, send 0

Router# show l2vpn atom vc
                                      Service
Interface Peer ID         VC ID      Type   Name                      Status
--------- --------------- ---------- ------ ------------------------- ----------
pw138     102.0.0.1       138        p2p    foo138                    UP

Router# show l2vpn atom vc detail

pseudowire138 is up, VC status is up PW type: Ethernet
  Create time: 00:00:23, last status change time: 00:00:20
    Last label FSM state change time: 00:00:20
  Destination address: 102.0.0.1 VC ID: 138
    Output interface: Fa0/0/1, imposed label stack {18 20}
    Preferred path: not configured
    Default path: active
    Next hop: 11.1.1.1
  Member of xconnect service foo138
    Associated member Gi1/0/0.10 is up, status is up
    Interworking type is Ethernet
    Service id: 0x7b000029
  Signaling protocol: LDP, peer 102.0.0.1:0 up
    Targeted Hello: 104.0.0.1(LDP Id) -> 102.0.0.1, LDP is UP
    Graceful restart: configured and enabled
    Non stop routing: not configured and not enabled
    PWid FEC (128), VC ID: 138
    Status TLV support (local/remote)         : enabled/supported
      LDP route watch                         : enabled
      Label/status state machine              : established, LruRru
      Local dataplane status received         : No fault
      BFD dataplane status received           : Not sent
      BFD peer monitor status received        : No fault
      Status received from access circuit     : No fault
      Status sent to access circuit           : No fault
      Status received from pseudowire i/f     : No fault
      Status sent to network peer             : No fault
      Status received from network peer       : No fault
      Adjacency status of remote peer         : No fault
  Sequencing: receive disabled, send disabled
  Bindings
    Parameter    Local                          Remote
    ------------ ------------------------------ ------------------------------
    Label        30                             20
    Group ID     0                              0
    Interface    Connect to CE2                 Connect to CE1
    MTU          1500                           1500
    Control word on (configured: autosense)     on
    PW type      Ethernet                       Ethernet
    VCCV CV type 0x02                           0x02
                   LSPV [2]                       LSPV [2]
    VCCV CC type 0x07                           0x07
                   CW [1], RA [2], TTL [3]        CW [1], RA [2], TTL [3]
    Status TLV   enabled                        supported
  SSO Descriptor: 102.0.0.1/138, local label: 30
```

```
  Dataplane:
    SSM segment/switch IDs: 4333/4332 (used), PWID: 41
  Rx Counters
    8 input transit packets, 312 bytes
    0 drops, 0 seq err
  Tx Counters
    5 output transit packets, 380 bytes
    0 drops
```

QinQ mode

The following example shows how to verify HDLC configuration on PE:

```
Router# show mpls l2transport vc

Local intf     Local circuit              Dest address    VC ID      Status
-------------  -------------------------- --------------- ---------- ----------
Se0/1/0:0      HDLC                       104.0.0.1       145        UP

Router# show mpls l2transport vc detail

Local interface: Se0/1/0:0 up, line protocol up, HDLC up
  Interworking type is Ethernet
  Destination address: 104.0.0.1, VC ID: 145, VC status: up
    Output interface: Fa0/0/1, imposed label stack {20 25}
    Preferred path: not configured
    Default path: active
    Next hop: 10.1.1.2
  Create time: 00:00:20, last status change time: 00:00:15
    Last label FSM state change time: 00:00:15
  Signaling protocol: LDP, peer 104.0.0.1:0 up
    Targeted Hello: 102.0.0.1(LDP Id) -> 104.0.0.1, LDP is UP
    Graceful restart: configured and enabled
    Non stop routing: not configured and not enabled
    Status TLV support (local/remote)   : enabled/supported
      LDP route watch                   : enabled
      Label/status state machine        : established, LruRru
      Last local dataplane   status rcvd: No fault
      Last BFD dataplane     status rcvd: Not sent
      Last BFD peer monitor  status rcvd: No fault
      Last local AC  circuit status rcvd: No fault
      Last local AC  circuit status sent: No fault
      Last local PW i/f circ status rcvd: No fault
      Last local LDP TLV     status sent: No fault
      Last remote LDP TLV    status rcvd: No fault
      Last remote LDP ADJ    status rcvd: No fault
    MPLS VC labels: local 27, remote 25
    Group ID: local 0, remote 0
    MTU: local 1500, remote 1500
    Remote interface description: Connect to CE2
  Sequencing: receive disabled, send disabled
  Control Word: On
  SSO Descriptor: 104.0.0.1/145, local label: 27
  Dataplane:
    SSM segment/switch IDs: 4521/4520 (used), PWID: 72
  VC statistics:
    transit packet totals: receive 4, send 7
    transit byte totals:   receive 216, send 427
    transit packet drops:  receive 0, seq error 0, send 0

Router# show l2vpn atom vc
                                     Service
Interface Peer ID         VC ID      Type   Name                     Status
```

```
--------- --------------- ---------- ------ ----------------------- ----------
pw145     104.0.0.1       145        p2p    foo145                  UP

Router# show l2vpn atom vc detail

pseudowire145 is up, VC status is up PW type: Ethernet
  Create time: 00:00:18, last status change time: 00:00:13
    Last label FSM state change time: 00:00:13
  Destination address: 104.0.0.1 VC ID: 145
    Output interface: Fa0/0/1, imposed label stack {16 33}
    Preferred path: not configured
    Default path: active
    Next hop: 10.1.1.2
  Member of xconnect service foo145
    Associated member Se0/1/0:0 is up, status is up
    Interworking type is Ethernet
    Service id: 0x2e
  Signaling protocol: LDP, peer 104.0.0.1:0 up
    Targeted Hello: 102.0.0.1(LDP Id) -> 104.0.0.1, LDP is UP
    Graceful restart: configured and enabled
    Non stop routing: not configured and not enabled
    PWid FEC (128), VC ID: 145
    Status TLV support (local/remote)         : enabled/supported
      LDP route watch                         : enabled
      Label/status state machine              : established, LruRru
      Local dataplane status received         : No fault
      BFD dataplane status received           : Not sent
      BFD peer monitor status received        : No fault
      Status received from access circuit     : No fault
      Status sent to access circuit           : No fault
      Status received from pseudowire i/f     : No fault
      Status sent to network peer             : No fault
      Status received from network peer       : No fault
      Adjacency status of remote peer         : No fault
  Sequencing: receive disabled, send disabled
  Bindings
    Parameter    Local                          Remote
    ------------ ------------------------------ ------------------------------
    Label        33                             33
    Group ID     0                              0
    Interface    Connect to CE1                 Connect to CE2
    MTU          1500                           1500
    Control word on (configured: autosense)     on
    PW type      Ethernet                       Ethernet
    VCCV CV type 0x02                           0x02
                   LSPV [2]                       LSPV [2]
    VCCV CC type 0x07                           0x07
                   CW [1], RA [2], TTL [3]        CW [1], RA [2], TTL [3]
    Status TLV   enabled                        supported
  SSO Descriptor: 104.0.0.1/145, local label: 33
  Dataplane:
    SSM segment/switch IDs: 4345/4344 (used), PWID: 48
  Rx Counters
    2 input transit packets, 108 bytes
    0 drops, 0 seq err
  Tx Counters
    3 output transit packets, 183 bytes
    0 drops
```

The following example shows how to verify the Qinq configuration on PE:

```
Router# show mpls l2transport vc
```

```
Local intf     Local circuit              Dest address    VC ID      Status
-------------  -------------------------- --------------- ---------- ----------
Gi1/0/0.10     Eth VLAN 10/20             102.0.0.1       145        UP

Router# show mpls l2transport vc detail

Local interface: Gi1/0/0.10 up, line protocol up, Eth VLAN 10/20 up
  Interworking type is Ethernet
  Destination address: 102.0.0.1, VC ID: 145, VC status: up
    Output interface: Fa0/0/1, imposed label stack {19 27}
    Preferred path: not configured
    Default path: active
    Next hop: 11.1.1.1
  Create time: 00:00:23, last status change time: 00:00:21
    Last label FSM state change time: 00:00:21
  Signaling protocol: LDP, peer 102.0.0.1:0 up
    Targeted Hello: 104.0.0.1(LDP Id) -> 102.0.0.1, LDP is UP
    Graceful restart: configured and enabled
    Non stop routing: not configured and not enabled
    Status TLV support (local/remote)   : enabled/supported
      LDP route watch                   : enabled
      Label/status state machine        : established, LruRru
      Last local dataplane   status rcvd: No fault
      Last BFD dataplane     status rcvd: Not sent
      Last BFD peer monitor  status rcvd: No fault
      Last local AC  circuit status rcvd: No fault
      Last local AC  circuit status sent: No fault
      Last local PW i/f circ status rcvd: No fault
      Last local LDP TLV     status sent: No fault
      Last remote LDP TLV    status rcvd: No fault
      Last remote LDP ADJ    status rcvd: No fault
    MPLS VC labels: local 25, remote 27
    Group ID: local 0, remote 0
    MTU: local 1500, remote 1500
    Remote interface description: Connect to CE1
  Sequencing: receive disabled, send disabled
  Control Word: On
  SSO Descriptor: 102.0.0.1/145, local label: 25
  Dataplane:
    SSM segment/switch IDs: 4815/4814 (used), PWID: 124
  VC statistics:
    transit packet totals: receive 10, send 6
    transit byte totals:   receive 430, send 456
    transit packet drops:  receive 0, seq error 0, send 0

Router# show l2vpn atom vc
                                      Service
Interface Peer ID         VC ID      Type   Name                      Status
--------- --------------- ---------- ------ ------------------------- ----------
pw145     102.0.0.1       145        p2p    foo145                    UP

Router# show l2vpn atom vc detail

pseudowire145 is up, VC status is up PW type: Ethernet
  Create time: 00:00:23, last status change time: 00:00:19
    Last label FSM state change time: 00:00:19
  Destination address: 102.0.0.1 VC ID: 145
    Output interface: Fa0/0/1, imposed label stack {18 33}
    Preferred path: not configured
    Default path: active
    Next hop: 11.1.1.1
  Member of xconnect service foo145
    Associated member Gi1/0/0.10 is up, status is up
    Interworking type is Ethernet
```

```
    Service id: 0xed000030
  Signaling protocol: LDP, peer 102.0.0.1:0 up
    Targeted Hello: 104.0.0.1(LDP Id) -> 102.0.0.1, LDP is UP
    Graceful restart: configured and enabled
    Non stop routing: not configured and not enabled
    PWid FEC (128), VC ID: 145
    Status TLV support (local/remote)         : enabled/supported
      LDP route watch                         : enabled
      Label/status state machine              : established, LruRru
      Local dataplane status received         : No fault
      BFD dataplane status received           : Not sent
      BFD peer monitor status received        : No fault
      Status received from access circuit     : No fault
      Status sent to access circuit           : No fault
      Status received from pseudowire i/f     : No fault
      Status sent to network peer             : No fault
      Status received from network peer       : No fault
      Adjacency status of remote peer         : No fault
  Sequencing: receive disabled, send disabled
  Bindings
    Parameter    Local                          Remote
    ------------ ------------------------------ ------------------------------
    Label        33                             33
    Group ID     0                              0
    Interface    Connect to CE2                 Connect to CE1
    MTU          1500                           1500
    Control word on (configured: autosense)     on
    PW type      Ethernet                       Ethernet
    VCCV CV type 0x02                           0x02
                   LSPV [2]                       LSPV [2]
    VCCV CC type 0x07                           0x07
                   CW [1], RA [2], TTL [3]        CW [1], RA [2], TTL [3]
    Status TLV   enabled                        supported
  SSO Descriptor: 102.0.0.1/145, local label: 33
  Dataplane:
    SSM segment/switch IDs: 4361/4360 (used), PWID: 48
  Rx Counters
    8 input transit packets, 344 bytes
    0 drops, 0 seq err
  Tx Counters
    5 output transit packets, 380 bytes
    0 drops
```

Example: HDLC-Dot1Q Interworking

The following example shows how to configure HDLC-dot1q interworking:

Short form

On HDLC-PE:

```
configure terminal
template type pseudowire hdlc-vlan1-tmp
 encapsulation mpls
 signaling protocol ldp

l2vpn xconnect context hdlc-vlan1
  interworking ethernet
  member Serial0/2/0:3
  member pseudowire101 3.3.3.3 107 template hdlc-vlan1-tmp
  no shutdown
end
```

On Ethernet-PE:

```
configure terminal
interface FastEthernet0/0/0.16
description Connect to CE2
encapsulation dot1q 16
no ip addres
no shut
!
template type pseudowire hdlc-vlan1-tmp
 encapsulation mpls
 signaling protocol ldp

l2vpn xconnect context hdlc-vlan1
  interworking ethernet
  member FastEthernet0/0/0.16
  member pseudowire101 1.1.1.1 107 template hdlc-vlan1-tmp
  no shutdown
end
```

Long form

On HDLC-PE:

```
configure terminal
template type pseudowire hdlc-vlan1
 encapsulation mpls
!
interface pseudowire107
 source template type pseudowire hdlc-vlan1
 encapsulation mpls
 neighbor 3.3.3.3 107
 signaling protocol ldp
 no shut
!
l2vpn xconnect context hdlc-vlan1-con
  interworking ethernet
  member Serial0/2/0:3
  member pseudowire107
  no shutdown
end
```

On Ethernet-PE:

```
configure terminal
interface FastEthernet0/0/0.16
description Connect to CE2
encapsulation dot1q 16
no ip addres
no shut
!
template type pseudowire hdlc-vlan1
 encapsulation mpls
!
interface pseudowire107
 source template type pseudowire hdlc-vlan1
 encapsulation mpls
 neighbor 1.1.1.1 107
 signaling protocol ldp
 no shut
!

l2vpn xconnect context hdlc-vlan1-con
  interworking ethernet
```

```
  member FastEthernet0/0/0.16
  member pseudowire107
  no shutdown
end
```

Additional References

The following sections provide references related to the Frame Relay-to-ATM Bridged Interworking and xconnect support on GEC (VPWS) features.

Related Documents

Related Topic	Document Title
Cisco IOS commands	*Cisco IOS Master Commands List, All Releases*
Basics of MPLS	*MPLS: Basic MPLS Configuration Guide, Cisco IOS XE Release 3S*
MPLS Layer 2 VPNs	*MPLS Layer 2 VPNs Configuration Guide, Cisco IOS XE Release 3S*
Cisco IOS Configuration Fundamentals	*Cisco IOS Configuration Fundamentals Command Reference*

Standards

Standard	Title
No new or modified standards are supported by this feature.	—

MIBs

MIB	MIBs Link
• CISCO-IETF-PW-MIB • CISCO-IETF-PW-MPLS-MIB	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use the Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

RFC[1]	Title
RFC 2684	Multiprotocol Encapsulation over ATM Adaptation Layer 5
RFC 2427	Multiprotocol Interconnect over Frame Relay

1. Not all the supported RFCs are listed.

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for Configuring MPLS Layer 2 VPNs

Table 16-2 lists the features in this module and provides links to specific configuration information. Only features that were introduced or modified in Cisco IOS Release 3.6.0S or a later release appear in the table.

Not all commands may be available in your Cisco IOS software release. For release information about a specific command, see the corresponding command reference documentation.

Use the Cisco Feature Navigator to find information about platform support and software image support. The Cisco Feature Navigator enables you to determine which Cisco IOS and Cisco Catalyst operating system software images support a specific software release, feature set, or platform. To access the Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 16-2 Feature Information for Configuring MPLS Layer 2 VPNs

Feature Name	Releases	Feature Information
Frame Relay to ATM Bridged Interworking	3.6.0S	The Frame Relay to ATM Bridged Interworking feature provides interoperability between the Frame Relay attachment VC and the ATM attachment VC connected to different PE routers. The bridged encapsulation corresponding to the bridged (Ethernet) interworking mechanism is used. The Ethernet frames are carried through the MPLS network using Ethernet over MPLS (EoMPLS). In Cisco IOS XE Release 3.6.0S, this feature was implemented on the Cisco ASR 1000 Series Aggregation Services Routers. The following sections provide information about this feature: • Frame Relay DLCI-to-ATM AAL5SNAP Bridged Interworking, page 16-13 • Configuring Frame Relay DLCI-to-ATM AAL5SNAP Bridged Interworking, page 16-15
xconnect support on GEC (VPWS) on ASR1000	3.6.0S	The Xconnect Support on GEC (VPWS) on ASR1000 feature enables the service providers to supply connectivity between customer sites with existing data link layer (Layer 2) networks by using a single, integrated, packet-based network infrastructure—a Cisco MPLS network. Instead of separate networks with network management environments, service providers can deliver Layer 2 connections over an MPLS backbone. In Cisco IOS XE Release 3.6.0S, this feature was implemented on the ASR 1000 Series Aggregation Services Routers. The following sections provide information about this feature: • Gigabit EtherChannel for Virtual Private Wire Service, page 16-26 • Configuring Gigabit EtherChannel for Virtual Private Wire Service, page 16-28

Table 16-2 Feature Information for Configuring MPLS Layer 2 VPNs

Feature Name	Releases	Feature Information
Reverse Layer 2 Gateway Protocol	3.8.0S	Reverse L2GP (R-L2GP) is a variation of L2GP. In case of R-L2GP, the pseudo information of the R-L2GP is transmitted by nPEs, instead of uPEs. R-L2GP provides a mechanism to send out static preconfigured BPDUs on each ring access port of nPEs to stimulate a per-access ring instantiation of the protocol. R-L2GP enables the PEs to avoid the burden of running Multiple-instance Spanning Tree Protocol (MST) when multiple independent access networks that run MST connect to a pair of redundant PEs. In order for this to work, the pair of nPEs are programmed to send out BPDUs on the access ring ports in such a way that they appear to be either: • The root bridge itself (the bridge with the lowest bridge id/priority). • The bridge with the second lowest bridge ID/priority, and with a 0 cost path to the root. The following sections provide information about this feature: • Reverse Layer 2 Gateway Protocol, page 16-5 • Configuring the R-L2GP, page 16-7
HDLC-Ethernet Interworking	3.13.0S	High-Level Data Link Control (HDLC)-Ethernet over MPLS is part of Any Transport over MPLS (AToM) solution. HDLC and Ethernet are two link-layer transport systems that utilize the AToM architecture. The feature describes how these two transport system can communicate with each other using the AToM framework. The following sections provide information about this feature: • High-Level Data Link Control-Ethernet Interworking, page 16-39 • Configuring HDLC-Ethernet Interworking, page 16-40

Glossary

ATM—Asynchronous Transfer Mode. A method of data transportation, whereby fixed-length packets are sent over a switched network. The method's ability to ensure reliable delivery of packets at a high rate makes it suitable for carrying voice, video, and data.

AToM—Any Transport over MPLS. AToM is a solution for transporting Layer 2 packets over an MPLS backbone. AToM enables service providers to supply connectivity between customer sites with existing data link layer (Layer 2) networks by using a single, integrated, packet-based network infrastructure—a Cisco MPLS network. Instead of separate networks with separate network management environments, service providers can deliver Layer 2 connections over an MPLS backbone.

Dot1q—IEEE 802.1Q is the networking standard that supports virtual LANs (VLANs) on an Ethernet network. The standard defines a system of VLAN tagging for Ethernet frames and the accompanying procedures to be used by bridges and switches in handling such frames.

EoMPLS—Ethernet over MPLS. This technology leverages an existing MPLS backbone network to deliver Transparent LAN Services based on Ethernet connectivity to the customer site.

GEC—Gigabit EtherChannel. A high-performance Ethernet technology that provides gigabit per second transmission rates. It provides a flexible and scalable bandwidth with resiliency and load sharing across links for switches, router interfaces, and servers. Supports up to eight links per channel.

HDLC—High-Level Data Link Control (HDLC) is a bit-oriented code-transparent synchronous data link-layer protocol developed by the International Organization for Standardization (ISO).

MPLS—Multiprotocol Label Switching. A mechanism in high-performance telecommunications networks that directs and carries data from one network node to the next. MPLS makes it easy to create virtual links between distant nodes. It can encapsulate packets of various network protocols.

QinQ—IEEE 802.1ad is an Ethernet networking standard informally known as IEEE 802.1QinQ, and is an amendment to the IEEE standard 802.1Q-1998. The technique is also known as provider bridging, Stacked VLANs, or simply QinQ or Q-in-Q.

VPLS—Virtual Private LAN Service. A method to provide Ethernet-based multipoint-to-multipoint communication over IP and MPLS networks.

Enabling Management by REST API

- Introduction
- Overview of Installation
- Enabling REST API Support Using the Cisco IOS XE CLI

Introduction

You can use the Cisco IOS XE REST API to manage the Cisco ASR 1001-X and ASR 1002-X as an alternative to configuring and managing selected features on the router using the Cisco IOS XE CLI. This chapter describes how to configure these Cisco ASR routers to enable management using the REST API. For detailed information about using the REST API, see the *Cisco IOS XE REST API Management Reference Guide*.

Overview of Installation

Installing the Cisco IOS XE REST API involves the following general steps:

1. Download the OVA package from Cisco.com.
   a. From the Cisco Routers product page, navigate to the Cisco CSR 1000V Cloud Services Router product page.
   http://www.cisco.com/c/en/us/products/routers/cloud-services-router-1000v-series/index.html
   b. Click the “Download Software” link.
   c. Select the Cisco IOS XE release package and follow the instructions for downloading the software.
2. Install the REST API OVA on the ASR platform.
3. Enable the REST API on the ASR platform. In the Enabling REST API Support Using the Cisco IOS XE CLI section, see:
   - Configuring the Dual Management Interface to Support the REST API
   - Configuring the REST API Local Port and AutoSave Options
   - Configuring onep

Enabling REST API Support Using the Cisco IOS XE CLI

- Configuring the Dual Management Interface to Support the REST API
- Configuring the REST API Local Port and AutoSave Options
- Configuring onep
- Disabling REST API Support
- Viewing the REST API Container Status

Configuring the Dual Management Interface to Support the REST API

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface GigabitEthernet***x*
4. **ip address** *ipv4-addr subnet-mask*
5. **no shutdown**
6. **exit**
7. **interface virtualportgroup** *virtual-port-group-number*
8. **ip unnumbered GigabitEthernet***x*
9. **no shutdown**
10. **exit**
11. **virtual-service csr_mgmt**
12. **vnic gateway virtualportgroup** *virtual-port-group-number*
13. **guest ip address** *remote-mgmt-ipv4-addr*
14. **exit**
15. **activate**
16. **end**
17. **ip route** *ip-address subnet-mask* **virtualportgroup** *virtual-port-group-number*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 3	**interface GigabitEthernet***x* **Example:** `Router(config)# interface gigabitethernet1`	Enters interface configuration mode for the interface designated by *x*. The range of GigabitEthernet ports depends on the platform.
Step 4	**ip address** *ipv4-addr subnet-mask* **Example:** `Router(config-if)# ip address 172.25.29.235 255.255.255.128`	Configures the IP address for the management interface.
Step 5	**no shutdown** **Example:** `Router(config-if)# no shutdown`	Enables the management interface.
Step 6	**exit** **Example:** `Router(config-if)# exit`	Exits interface configuration mode.
Step 7	**interface virtualportgroup** *virtualportgroup-number* **Example:** `Router(config)# interface virtualportgroup 0`	Creates a virtual port group and enters virtual port group interface configuration mode.
Step 8	**ip unnumbered GigabitEthernet***x* **Example:** `router(config-if)# ip unnumbered gigabitethernet1`	Enables IP processing on an interface without assigning it an explicit IP address.
Step 9	**no shutdown** **Example:** `router(config-if)# no shutdown`	Enables the virtual port group interface.
Step 10	**exit** **Example:** `router(config-if)# exit`	Exits virtual port group interface mode.
Step 11	**virtual-service csr_mgmt** **Example:** `router(config)# virtual-service csr_mgmt`	Configures the virtual services container and enters virtual services configuration mode.
Step 12	**vnic gateway virtualportgroup** *virtualportgroup_number* **Example:** `router(config-virt-serv)# vnic gateway virtualportgroup 0`	Creates a vNIC gateway interface for the virtual services container and maps it to the virtual port group.

	Command or Action	Purpose
Step 13	**guest ip address** *remote-mgmt-ipv4-addr* **Example:** `router(config-virt-serv-intf)# guest ip address 172.25.29.236`	Configures the remote-management IP address for the vNIC gateway interface for the virtual services container.
Step 14	**exit** **Example:** `router(config-virt-serv-intf)# exit`	Exits virtual services interface configuration mode and enters virtual services configuration mode.
Step 15	**activate** **Example:** `router(config-virt-serv)# activate`	Activates the **csr_mgmt** virtual services container.
Step 16	**end** **Example:** `router(config-virt-serv)# end`	Exits virtual services configuration mode and enters global configuration mode.
Step 17	**ip route** *ipaddress subnetmask* **virtualportgroup** *virtualport-groupnumber* **Example:** `router(config)# ip route 172.25.29.236 255.255.255.255 VirtualPortGroup0`	Creates an IP route that maps to the virtual port group. Use the same IP address that was configured using the **guest ip address** command.

Configuring the REST API Local Port and AutoSave Options

Beginning with Cisco IOS XE 3.13S, you can configure the REST API local port and autosave options.

SUMMARY STEPS

1. **remote-management**
2. **restful-api local port** *local-port-number*
3. **restful-api autosave** *interval*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**remote-management** **Example:** `router(config)# remote-management`	Enters remote-management configuration mode.
Step 2	**restful-api local-port** *local-port-number* **Example:** `router(cfg-remote-mgmt)# restful-api local-port 55433`	Configures the REST API local port number. The valid range depends on whether the REST API virtual services container uses the same IP address as the management interface, or if it uses a different IP address: • Valid range if the dual management interface is configured is from 1 to 61000. • Valid range if the shared management interface is configured is from 55001 to 61000. In both cases, the default value is 55433.
Step 3	**restful-api autosave** *interval* **Example:** `Router(cfg-remote-mgmt)# restful-api autosave 60`	Configures the REST API autosave interval. The range is from 30-300 seconds, and the default is 30.

Configuring onep

The Open Network Environment Programming Interface (onep) is used to define the service set for Cisco IOS and the REST API. Configure onep as follows.

SUMMARY STEPS

1. **conf t**
2. **onep**
3. **service set vty**
4. **end**

DETAILED STEPS

	Command or Action	Purpose

Step 1	**conf t**
	
	Example:
	`asr1k#conf t`
	`Enter configuration commands, one per line.  End with CNTL/Z.`
		Enters configuration mode.

Step 2	**onep** **Example:** `asr1k(config)#onep`	Enters onep mode.
Step 3	**service set vty** **Example:** `asr1k(config-onep)#service set vty`	Select the vty service set.

Step 4	**end**
	
	Example:
	`asr1k(config-onep)#end`
	`asr1k#`
		Exit onep mode.

Disabling REST API Support

Support for the REST API is enabled by default. The following procedure disables the REST API.

SUMMARY STEPS

1. **enable**
2. **configure terminal**

3. **remote-management**
4. **no restful-api**
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `router# configure terminal`	Enters global configuration mode.
Step 3	**remote-management** **Example:** `router(config)# remote-management`	Enters remote-management configuration mode.
Step 4	**no restful-api** **Example:** `router(cfg-remote-mgmt)# no restful-api`	Disables support for the REST API.
Step 5	**end** **Example:** `router(cfg-remote-mgmt)# end`	Exits remote-management configuration mode and enters configuration mode.

Note When REST API support is disabled using the **no restful-api** command, the REST API PUT, POST and DELETE operations are disabled. However, the GET operation is still available.

Viewing the REST API Container Status

Use the **show virtual-service detail** command to view the REST API container status.

CHAPTER 18

LSM-MLDP-based MVPN Support

First Published: November 28, 2012

The Label Switched Multicast (LSM) feature supports IPv4 and IPv6 multicast traffic over a Multi-Protocol Label Switching (MPLS) network. This feature is based on the basic MPLS infrastructure and supports IP multicast traffic through the MPLS clouds. The LSM feature enables service providers to extend the existing MPLS backbone network for multicast services. By default, MPLS creates an out-label for an in-label for each packet. This feature extends this functionality to create multiple out-labels for a single in-label.

The LSM service includes point-to-multipoint (P2MP) and multipoint-to-multipoint (MP2MP) packet transport. The P2MP packet transport can be implemented using either Resource reSerVation Protocol (RSVP) P2MP - Traffic Engineering (P2MP-TE), or Multicast Label Distribution Protocol (MLDP) based Multicast VPN (MVPN). The MP2MP packet transport can be implemented only through MLDP based MVPN.

The packets are transported over three types of routers:

- Head-end router: Encapsulates the IP packet with one or more labels.
- Midpoint router: Replaces the in-label with an out-label.
- Tail-end router: Removes the label from the packet.

Restrictions and Usage Guidelines

Follow these restrictions and usage guidelines while configuring LSM-MLDP-based MVPN support:

- A head-end router does not support multiple sub Label Switched Paths (subLSPs) belonging to different tunnels, over the same physical interface.
- RSVP-TE-based LSM is not supported; only MLDP-based LSM is supported.
- Process-level software forwarding is not supported.
- Rosen Model MLDP is not supported in the global configuration mode. However, MLDP inband signaling is supported in the global configuration mode.
- These are the scale considerations for MLDP-based MVPN:
  - Maximum number of Multicast Virtual Route Forwardings (MVRFs) supported on each PE is 600.
  - Maximum number of m-route supported on each PE is 200,000.
  - Maximum number of OIF supported is 1000.

  - Maximum number of MLDP ingress labels (local labels) supported on each PE is 100,000.
  - Maximum number of MLDP egress labels (remote labels) supported on each PE is 100,000.
  - Max of 32 PE or P neighbors in a PE router per MDT, and max of 33 PE or P neighbors in a P router per MDT.
- Supported content group modes are Protocol Independent Multicast (PIM) sparse mode (PIM-SM) and Source Specific Multicast (SSM) traffic.
- Unsupported content group modes are PIM dense mode (PIM-DM) and bidirectional PIM (bidir-PIM) traffic.
- The PIM-sparse content group mode is supported if the RP is configured behind the PE router (on CE). The RP and the source router have to be in the same VRF and PE site with the same RPF interface.
- For RPF lookup in the context of the extranet, only the **ip multicast rpf select** command is supported for the configuration.
- The MLDP provides only link protection with the FRR TE. Only single hop is supported with MLDP TE. However, the backup path can have multiple hops.

Configuring LSM-MLDP-based MVPN Support

Deployment of an LSM-MLDP-based MVPN involves configuring a default Multicast Distribution Trees (MDT) and one or more data MDTs.

A static default MDT is established for each multicast domain. The default MDT defines the path used by PE routers to send multicast data and control messages to other PE routers in the multicast domain. A default MDT is created in the core network using a single MP2MP LSP.

An MLDP-based MVPN also supports dynamic creation of data MDTs for high-bandwidth transmissions. For high-rate data sources, a data MDT is created using the P2MP LSPs to offload the traffic from the default MDT to avoid unnecessary wastage of bandwidth to PEs that are not a part of the stream. You can configure MLDP MVPN for both the intranet and the extranet.

Note Before configuring MLDP-based MVPN, ensure that the MPLS is enabled on the core facing interface. For information on MPLS configuration, see the *Cisco IOS Multiprotocol Label Switching Configuration Guide*. Also, ensure that the BGP and any interior gateway protocol (OSPF or ISIS) is enabled on the core router.

Configuring MLDP MVPN Intranet Services

Complete these steps to configure MLDP MVPN for intranet:

- Enabling MPLS MLDP
- Configuring MVPN Routing and Forwarding instance
- Configuring a VRF entry
- Configuring the route distinguisher
- Configuring VPN Id
- Configuring the Route-Target extended community
- Configuring the default MDT

- Configuring Data MDTs (optional)
- Configuring BGP MDT address family
- Configuring BGP vpnv4 address family
- Configuring BGP VRF address family
- Configuring PIM SM/SSM mode for the VRFs

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **mpls MLDP**
4. **vrf definition** *vrf-name*
5. **rd** *route-distinguisher*
6. **vpn id** *vpn_id*
7. **route-target** [**import** | **export** | **both**] *route-target-ext-community*
8. **route-target [import | export | both**] *route-target-ext-community*
9. **mdt default mpls MLDP** *root-node*
10. **mdt data mpls MLDP** *numberofdataMDTs*
11. **mdt data threshold** *bandwidth*
12. **exit**
13. **ip multicast-routing vrf** *vrf-name* **distributed**
14. **end**

Note See Configuring the MDT Address Family in BGP for Multicast VPN for information on configuring an MDT and vpnv4 address family session on the PE routers to establish MDT peering sessions for MVPN.

DETAILED STEPS

	Command	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password when prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.

	Command	Purpose
Step 3	**mpls MLDP** **Example:** `Router(config)# mpls MLDP`	Enables MPLS MLDP support. **Note** The **mpls MLDP** command is configured by default. To disable MPLS MLDP, use the **no mpls MLDP** command.
Step 4	**vrf definition** *vrf-name* **Example:** `Router(config)# ip vrf blue`	Defines the VPN routing instance by assigning a VRF name, and enters the VRF configuration mode. The *vrf-name* argument is the name assigned to a VRF.
Step 5	**rd** *route-distinguisher* **Example:** `Router(config-vrf)# rd 10:3`	Creates routing and forwarding tables. Specify the *route-distinguisher* argument to add an 8-byte value to create a VPN prefix. You can enter an *route-distinguisher* value in either of these formats: • 16-bit autonomous system number: Your 16-bit number. For example, 101:3. • 32-bit IP address: Your 32-bit number. For example, 192.168.122.15:1.
Step 6	**vpn id** *vpn-id* **Example:** `Router(config-vrf)# vpn id 10:3`	Sets or updates a VPN identifier on a VRF.
Step 7	**route-target import** *route-target-ext-community* **Example:** `Router(config-vrf)# route-target import 10:3`	Creates a route-target extended community for a VRF. • The import keyword imports the routing information from the target VPN extended community. • The route-target-ext-community argument adds the route-target extended community attributes to the VRF list of import, export, or both (import and export) route-target extended communities.
Step 8	**route-target export** *route-target-ext-community* **Example:** `Router(config-vrf)# route-target export 10:3`	Creates a route-target extended community for a VRF. • The export keyword exports the routing information from the target VPN extended community. • The route-target-ext-community argument adds the route-target extended community attributes to the VRF list of import, export, or both (import and export) route-target extended communities.

	Command	Purpose
Step 9	**mdt default mpls MLDP** *root-node* **Example:** `Router(config-vrf)# mdt default mpls MLDP 2.2.2.2`	Configures MLDP MDT for a VRF. The root node can be IP address of a loopback or physical interface on any router (source PE, receiver PE or core router) in the provider network. The root node address should be reachable by all the routers in the network. The router from where the signalling occurs functions as the root node. The default MDT must be configured on each PE router to enable the PE routers to receive multicast traffic for this particular MVRF. **Note** By default MPLS MLDP is enabled. To disable, use the **no mpls MLDP** command. **Note** LSPVIF tunnel is created as a result of **mdt default mpls MLDP** *root-node* command.
Step 10	**mdt data mpls MLDP** *numberofdataMDTs* **Example:** `Router(config-vrf)# mdt data mpls MLDP 100`	Configures the MLDP data MDP.
Step 11	**mdt data threshold** *bandwidth* **Example:** `Router(config-vrf)# mdt data threshold 20`	Configures the threshold value for data MDT. **Note** Bandwidth is traffic rate in Kb/s.
Step 12	**exit** **Example:** `Router(config-vrf)# exit`	Exits the configuration session.
Step 13	**ip multicast-routing vrf** *vrf-name* distributed **Example:** `Router(config)# ip multicast-routing vrf blue distributed`	Enables multicast routing for the specified VRF.
Step 14	**end** **Example:** `Router(config)# end`	Closes the configuration session.

Note See Configuring the MDT Address Family in BGP for Multicast VPN for information on configuring an MDT address family session on the PE routers to establish MDT peering sessions for MVPN.

Example

This example describes how to configure MLDP MVPN on an intranet:

```
Router> enable
Router# configure terminal
Router(config)# mpls MLDP
Router(config)# ip vrf blue
Router(config-vrf)# rd 10:3
Router(config-vrf)# vpn id 10:3
Router(config-vrf)# route-target import 10:3
Router(config-vrf)# route-target export 10:3
Router(config-vrf)# mdt default mpls MLDP 2.2.2.2
Router(config-vrf)# mdt data mpls MLDP 100
Router(config-vrf)# mdt data threshold 20
Router(config-vrf)# exit
Router(config)# ip multicast-routing vrf blue distributed
Router(config)# end
```

Verification

Use these commands to verify the LSM-MLDP-based MVPN support intranet configuration.

- To check the MLDP neighbors, use the **show mpls MLDP neighbors** command:

```
Router# show mpls MLDP neighbors
MLDP peer ID    : 3.3.3.3:0, uptime 00:41:41 Up,
  Target Adj     : Yes
  Session hndl   : 2
  Upstream count : 2
  Branch count   : 0
  Path count     : 1
  Path(s)        : 3.3.3.3           No LDP Tunnel20
  Nhop count     : 1
  Nhop list      : 3.3.3.3

 MLDP peer ID    : 2.2.2.2:0, uptime 00:17:42 Up,
  Target Adj     : No
  Session hndl   : 4
  Upstream count : 0
  Branch count   : 0
  Path count     : 1
  Path(s)        : 3.3.3.3           No LDP Tunnel20
  Nhop count     : 0
```

- To check the PIM neighbors, use the **show ip pim vrf** *vrf-name* **neighbor** command:

```
Router# show ip pim vrf blue neighbor
PIM Neighbor Table
Mode: B - Bidir Capable, DR - Designated Router, N - Default DR Priority,
      P - Proxy Capable, S - State Refresh Capable, G - GenID Capable
Neighbor          Interface               Uptime/Expires    Ver   DR
Address                                                           Prio/Mode
3.3.3.3           Lspvif1                 00:06:21/00:01:17 v2    1 / DR S P G
```

- To check the multicast routes for a given VRF, use **show ip mroute vrf** *vrf_name* **verbose** command:

```
Router# show ip mroute vrf blue verbose
```

```
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
       L - Local, P - Pruned, R - RP-bit set, F - Register flag,
       T - SPT-bit set, J - Join SPT, M - MSDP created entry, E - Extranet,
       X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
       U - URD, I - Received Source Specific Host Report,
       Z - Multicast Tunnel, z - MDT-data group sender,
       Y - Joined MDT-data group, y - Sending to MDT-data group,
       V - RD & Vector, v - Vector
Outgoing interface flags: H - Hardware switched, A - Assert winner
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode

(40.0.0.2, 232.0.1.4), 00:00:16/00:03:13, flags: sT
  Incoming interface: GigabitEthernet3/2/1, RPF nbr 0.0.0.0
  Outgoing interface list:
    Lspvif1, LSM MDT: B0000004 (default), Forward/Sparse, 00:00:16/00:03:13

(*, 224.0.1.40), 00:47:09/00:02:56, RP 0.0.0.0, flags: DPL
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list: Null
```

- To check the packet counters, use **show ip mroute vrf** *vrf_name* **count** command:

```
Router# show ip mroute vrf blue count
IP Multicast Statistics
2 routes using 1208 bytes of memory
2 groups, 0.50 average sources per group
Forwarding Counts: Pkt Count/Pkts per second/Avg Pkt Size/Kilobits per second
Other counts: Total/RPF failed/Other drops(OIF-null, rate-limit etc)

Group: 232.0.1.4, Source count: 1, Packets forwarded: 1333, Packets received: 1334
  Source: 40.0.0.2/32, Forwarding: 1333/20/46/7, Other: 1334/0/1

Group: 224.0.1.40, Source count: 0, Packets forwarded: 0, Packets received: 0
```

- To check the MPLS forwarding, use **show mpls forwarding-table** command:

```
Router# show mpls forwarding-table
Local Outgoing Prefix Bytes Label Outgoing Next Hop
Label      Label      or Tunnel Id     Switched      interface
16 Pop Label IPv4 VRF[V] 0 aggregate/blue
17 Pop Label  IPv4 VRF[V] 0 aggregate/red
18 [T] Pop Label  3.3.3.3/32 0 Tu20 point2point
19 [T]  25 2.2.2.2/32 0 Tu20 point2point
20 [T]  Pop Label 19.0.0.0/24 0 Tu20 point2point
22 [T]  No Label [mdt 55:1111 0][V] \9422 aggregate/red
23 [T]  No Label [mdt 55:2222 0][V] \9708          aggregate/blue
[T]     Forwarding through a LSP tunnel.
        View additional labelling info with the 'detail' option
```

Configuring MLDP MVPN for Extranet Services

You can configure MLDP MVPN for extranet services using these methods:

- Source-Side Chaining (SSC): Configure the phantom receiver MVRF on the source-side router. Multicast routes with VRF Reverse Path Forwarding (RPF) loopup should be configured on the source PE.
- Receiver-Side Chaining (RSC): Configure the phantom source MVRF on the receiver-side router. Multicast routes with VRF RPF loopup should be configured on the receiver VRF.

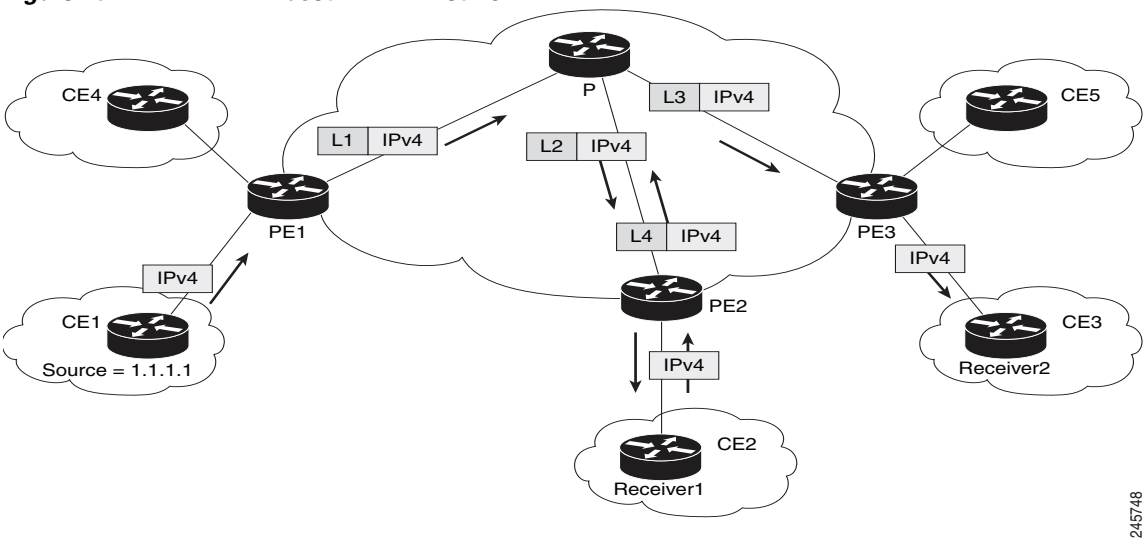
Configuring MLDP MVPN for Extranet using SSC

Complete these steps to configure the MLDP MVPN extranet support using SSC:

- Configuring receiver MVRF on the source PE.
- Configuring a loopback address in the receiver VRF on the source PE.
- Configuring fallback multicast route for source address on source PE.
- Configuring fallback multicast route for RP address on the source PE in case of SM mode.
- Configuring static multicast route on recevier PE for loopback IP in the receiver VRF configured on the source PE.

Note This configuration is based on illustration Figure 18-1. Configure multicast routes on the PE1 router.

Figure 18-1 MLDP Based MVPN Network



SUMMARY STEPS

The followings are the summary steps to configure MLDP MVPN for Extranet using SSC.

Configuration on the Source PE:

1. **enable**
2. **configure terminal**
3. **vrf definition** *vrf-name*
4. **rd** *route-distinguisher*
5. **vpn id** *vpn_id*
6. **route-target import** *route-target-ext-community*
7. **route-target import** *route-target-ext-community*
8. **mdt default mpls MLDP** *root-node*
9. **end**

10. **interface type** *instance*
11. **ip vrf forwarding** *vrf-name*
12. **ip address** *ip-address subnet*
13. **exit**
14. **ip multicast** [**vrf** *receiver-vrf-name*] **rpf select** {**global** | *vrf source-vrf-name*} **group-list** *access-list*
15. **end**

Configuration on the receiver PE:

1. **enable**
2. **configure terminal**
3. **vrf definition** *vrf-name*
4. **rd** *route-distinguisher*
5. **vpn id** *vpn_id*
6. **route-target import** *route-target-ext-community*
7. **route-target import** *route-target-ext-community*
8. **mdt default mpls MLDP** *root-node*
9. **end**
10. **interface type** *instance*
11. **ip vrf forwarding** *vrf-name*
12. **ip address** *ip-address subnet*
13. **exit**
14. **ip mroute vrf** *receiver_vrf source_address subnet_mask loopback_ip*
15. **end**

DETAILED STEPS

The followings are the detailed steps to configure MLDP MVPN for Extranet using SSC.

Configuration on the Source PE:

	Command	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password when prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.

	Command	Purpose
Step 3	**vrf definition** *vrf-name* **Example:** `Router(config)# vrf definition blue`	Defines the VPN routing instance by assigning a *VRF name* argument, and enters the VRF configuration mode. The vrf-name argument is the name assigned to a VRF.
Step 4	**rd** *route-distinguisher* **Example:** `Router(config-if)# rd 10:4`	Creates routing and forwarding tables. Specify the *route-distinguisher* argument to add an 8-byte value to create a VPN prefix. You can enter an *route-distinguisher* value in either of these formats: • 16-bit autonomous system number: Your 16-bit number. For example, 101:3. • 32-bit IP address: Your 32-bit number. For example, 192.168.122.15:1.
Step 5	**vpn id** *vpn-id* **Example:** `Router(config-if)# vpn id 10:4`	Sets or updates a VPN identifier on a VRF.
Step 6	**route-target import** *route-target-ext-community* **Example:** `Router(config-vrf)# route-target import 10:4`	Creates a route-target extended community for a VRF. • The import keyword imports the routing information from the target VPN extended community. • The route-target-ext-community argument adds the route-target extended community attributes to the VRF list of import, export, or both (import and export) route-target extended communities.
Step 7	**route-target export** *route-target-ext-community* **Example:** `Router(config-vrf)# route-target export 10:4`	Creates a route-target extended community for a VRF. • The export keyword export the routing information to the target VPN extended community. • The route-target-ext-community argument adds the route-target extended community attributes to the VRF list of import, export, or both (import and export) route-target extended communities.
Step 8	**mdt default mpls MLDP** *root-node* **Example:** `Router(config-vrf)# mdt default mpls MLDP 2.2.2.2`	Configures MLDP multicast distribution tree (MDT) for a VRF. **Note** LSPVIF tunnel is created as a result of this command.
Step 9	**end** **Example:** `Router(config-vrf)# end`	Closes the configuration session.

	Command	Purpose
Step 10	**interface** *type instance* **Example:** `Router(config)# interface loopback 3`	Enters interface configuration mode and names the new loopback interface.
Step 11	**ip vrf forwarding** *vrf-name* **Example:** `Router(config-if)# ip vrf forwarding red`	Associates a VRF instance with an interface or subinterface. • vrf-name is the name assigned to a VRF.
Step 12	**ip address** *ip-address subnet-mask* **Example:** `Router(config-if)# ip address 1.1.1.1 255.255.255.255`	Specifies the interface IP address and subnet mask. • ip-address specifies the IP address of the interface. • subnet-mask specifies the subnet mask of the interface.
Step 13	**exit**	Exits the interface configuration mode.
Step 14	**ip multicast** [**vrf** receiver-vrf-name] **rpf select** {**global** | **vrf** source-vrf-name} **group-list** access-list **Example:** `Router(config)# ip multicast vrf red rpf select vrf blue`	Configures Reverse Path Forwarding (RPF) lookups originating in a receiver Multicast VPN (MVPN) routing and forwarding (MVRF) instance, in the global routing table to be performed in a source MVRF instance, or in the global routing table based on group address. The optional **vrf** keyword and *receiver-vrf-name* argument are used to apply a group-based VRF selection policy to RPF lookups originating in the VRF specified for the *receiver-vrf-name* argument. If the optional **vrf** keyword and *receiver-vrf-name* argument are not specified, the group-based VRF selection policy applies to RPF lookups originating from the global table.
Step 15	**end** **Example:** `Router(config-vrf)# end`	Closes the configuration session.

Configuration on Receiver PE:

	Command	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password when prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.

	Command	Purpose
Step 3	**vrf definition** *vrf-name* **Example:** `Router(config)# vrf definition blue`	Defines the VPN routing instance by assigning a VRF name, and enters the VRF configuration mode. The *vrf-name* argument is the name assigned to a VRF.
Step 4	**rd** *route-distinguisher* **Example:** `Router(config-if)# rd 10:4`	Creates routing and forwarding tables. Specify the *route-distinguisher* argument to add an 8-byte value to create a VPN prefix. You can enter an RD value in either of these formats: • 16-bit autonomous system number: Your 16-bit number. For example, 101:3. • 32-bit IP address: Your 32-bit IP address. For example, 192.168.122.15:1.
Step 5	**vpn id** *vpn-id* **Example:** `Router(config-if)# vpn id 10:4`	Sets or updates a VPN identifier on a VRF.
Step 6	**route-target import** *route-target-ext-community* **Example:** `Router(config-vrf)# route-target import 10:4`	Creates a route-target extended community for a VRF. • The import keyword imports the routing information from the target VPN extended community. • The route-target-ext-community argument adds the route-target extended community attributes to the VRF list of import, export, or both (import and export) route-target extended communities.
Step 7	**route-target export** *route-target-ext-community* **Example:** `Router(config-vrf)# route-target export 10:4`	Creates a route-target extended community for a VRF. • The export keyword exports the routing information to the target VPN extended community. • The route-target-ext-community argument adds the route-target extended community attributes to the VRF list of import, export, or both (import and export) route-target extended communities.
Step 8	**mdt default mpls MLDP** *root-node* **Example:** `Router(config-vrf)# mdt default mpls MLDP 2.2.2.2`	Configures MLDP multicast distribution tree (MDT) for a VRF. **Note** LSPVIF tunnel is created as a result of this command.
Step 9	**end** **Example:** `Router(config-vrf)# end`	Closes the configuration session.
Step 10	**interface** *type instance* **Example:** `Router(config)# interface loopback 3`	Enters interface configuration mode and names the new loopback interface.

	Command	Purpose
Step 11	**ip vrf forwarding** *vrf-name* **Example:** `Router(config-if)# ip vrf forwarding blue`	Associates a VRF instance with an interface or subinterface. • vrf-name is the name assigned to a VRF.
Step 12	**ip address** *ip-address subnet-mask* **Example:** `Router(config-if)# ip address 3.3.3.3 255.255.255.255`	Specifies the interface IP address and subnet mask. • ip-address specifies the IP address of the interface. • subnet-mask specifies the subnet mask of the interface.
Step 13	**exit**	Exits the interface configuration mode.
Step 14	**ip mroute vrf** *receiver_vrf source_address subnet_mask loopback_ip* **Example:** `Router(config-if)# ip mroute vrf red 40.0.0.0 255.255.255.0 1.1.1.1`	Configures the static multicast routes for source addresses in the reciever VRF, where: loopback ip is ip address of the loopback configured in the recevier VRF in the source PE.
Step 15	**end** **Example:** `Router(config-vrf)# end`	Closes the configuration session.

Example

This is sample example for configuring MLDP MVPN for configuring extranet using SSC:

Configuration on the Source PE(Configure These Steps for Both Red and Blue VRFs)

```
Router> enable
Router# configure terminal
Router(config)# ip vrf blue
Router(config-if)# rd 10:4
Router(config-if)# vpn id 10:4
Router(config-vrf)# route-target import 10:4
Router(config-vrf)# route-target export 10:4
Router(config-vrf)# mdt default mpls MLDP 2.2.2.2
Router(config-vrf)# end
Router(config)# interface loopback 3
Router(config-if)# ip vrf forwarding red
Router(config-if)# ip address 1.1.1.1 255.255.255.255
Router(config)# ip mroute vrf red 40.0.0.0 255.255.255.0 fallback-lookup vrf blue
Router(config)# ip mroute vrf red 44.44.44.44 255.255.255.0 fallback-lookup vrf blue
Router(config-vrf)# end
```

Configuration on the Receiver PE

```
Router> enable
Router# configure terminal
Router(config)# ip vrf blue
Router(config-if)# rd 10:4
Router(config-if)# vpn id 10:4
Router(config-vrf)# route-target import 10:4
```

```
Router(config-vrf)# route-target export 10:4
Router(config-vrf)# mdt default mpls MLDP 2.2.2.2
Router(config-vrf)# end
Router(config)# interface loopback 3
Router(config-if)# ip vrf forwarding blue
Router(config-if)# ip address 3.3.3.3 255.255.255.255 Remove
Router(config-if)# ip mroute vrf red 40.0.0.0 255.255.255.0 1.1.1.1
Router(config-vrf)# end
```

Configuring MLDP MVPN for Extranet Services using RSC

Complete these steps to configuring MLDP MVPN for extranet services using RSC:

- Configuring the source mVRF on the receiver PE router.
- Configuring RPF for MLDP based MVPN extranet support using static multicast routes on the receiver PE.

Note This configuration is based on illustration Figure 18-1. Configure multicast routes on PE2 and PE3 routers.

SUMMARY STEPS

Configuration on Source PE:

1. **enable**
2. **configure terminal**
3. **vrf definition** *vrf-name*
4. **rd** *route-distinguisher*
5. **vpn id** *vpn_id*
6. **route-target import** *route-target-ext-community*
7. **route-target import** *route-target-ext-community*
8. **mdt default mpls MLDP** *root-node*
9. **end**

Configuration on Receiver PE (Configure these steps for both red and blue VRFs)

1. **enable**
2. **configure terminal**
3. **vrf definition** *vrf-name*
4. **rd** *route-distinguisher*
5. **vpn id** *vpn_id*
6. **route-target import** *route-target-ext-community*
7. **route-target import** *route-target-ext-community*
8. **mdt default mpls MLDP** *root-node*
9. **ip mroute** [**vrf** *recevier-vrf-name*] *source-address mask* **fallback-lookup** {**global** | **vrf** *source-vrf-name*} [*distance*]
10. **end**

DETAILED STEPS

Configuration on Source PE

	Command	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password when prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**vrf definition** *vrf-name* **Example:** `Router(config)# ip vrf blue`	Defines the VPN routing instance by assigning a VRF name, and enters the VRF configuration mode. The *vrf-name* argument is the name assigned to a VRF.
Step 4	**rd** *route-distinguisher* **Example:** `Router(config-if)# rd 10:3`	Creates routing and forwarding tables. Specify the *route-distinguisher* argument to add an 8-byte value to create a VPN prefix. You can enter an RD value in either of these formats: • 16-bit autonomous system number: Your 16-bit number. For example, 101:3. • 32-bit IP address: Your 32-bit number. For example, 192.168.122.15:1.
Step 5	**vpn id** *vpn-id* **Example:** `Router(config-if)# vpn id 10:3`	Sets or updates a VPN identifier on a VRF.
Step 6	**route-target import** *route-target-ext-community* **Example:** `Router(config-vrf)# route-target import 10:3`	Creates a route-target extended community for a VRF. • The import keyword imports routing information from the target VPN extended community. • The route-target-ext-community argument adds the route-target extended community attributes to the VRF list of import, export, or both (import and export) route-target extended communities.
Step 7	**route-target export** *route-target-ext-community* **Example:** `Router(config-vrf)# route-target export 10:3`	Creates a route-target extended community for a VRF. • The export keyword exports the routing information to the target VPN extended community. • The route-target-ext-community argument adds the route-target extended community attributes to the VRF list of import, export, or both (import and export) route-target extended communities.

	Command	Purpose
Step 8	**mdt default mpls MLDP** *root-node* **Example:** `Router(config-vrf)# mdt default mpls MLDP 2.2.2.2`	Configures MLDP multicast distribution tree (MDT) for a VRF. **Note** LSPVIF tunnel is created as a result of this command.
Step 9	**end** **Example:** `Router(config-vrf)# end`	Closes the configuration session.

Configuration on Receiver PE

	Command	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password when prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**vrf definition** *vrf-name* **Example:** `Router(config)# ip vrf blue`	Defines the VPN routing instance by assigning a VRF name, and enters the VRF configuration mode. The *vrf-name* argument is the name assigned to a VRF.
Step 4	**rd** *route-distinguisher* **Example:** `Router(config-if)# rd 10:3`	Creates routing and forwarding tables. Specify the *route-distinguisher* argument to add an 8-byte value to create a VPN prefix. You can enter an RD value in either of these formats: • 16-bit autonomous system number: Your 16-bit number. For example, 101:3. • 32-bit IP address: Your 32-bit number. For example, 192.168.122.15:1.
Step 5	**vpn id** *vpn-id* **Example:** `Router(config-if)# vpn id 10:3`	Sets or updates a VPN identifier on a VRF.

	Command	Purpose
Step 6	**route-target import** *route-target-ext-community* **Example:** `Router(config-vrf)# route-target import 10:3`	Creates a route-target extended community for a VRF. • The import keyword imports routing information from the target VPN extended community. • The route-target-ext-community argument adds the route-target extended community attributes to the VRF list of import, export, or both (import and export) route-target extended communities.
Step 7	**route-target export** *route-target-ext-community* **Example:** `Router(config-vrf)# route-target export 10:3`	Creates a route-target extended community for a VRF. • The export keyword exports the routing information to the target VPN extended community. • The route-target-ext-community argument adds the route-target extended community attributes to the VRF list of import, export, or both (import and export) route-target extended communities.
Step 8	**mdt default mpls MLDP** *root-node* **Example:** `Router(config-vrf)# mdt default mpls MLDP 2.2.2.2`	Configures MLDP multicast distribution tree (MDT) for a VRF. **Note** LSPVIF tunnel is created as a result of this command.
Step 9	**ip mroute** [**vrf** receiver-vrf-name] source-address mask {**fallback-lookup vrf** source-vrf-name} [*distance*] **Example:** `Router(config)# ip mroute vrf red 40.0.0.0 255.255.255.0 fallback-lookup vrf blue`	Configures RPF lookups originating in a receiver MVRF or in the global routing table to be resolved in a source MVRF or in the global routing table based on group address. Use this command on the receiver PE. • The optional vrf keyword and receiver-vrf-name argument are used to apply a group-based VRF selection policy to RPF lookups originating in the VRF specified for the receiver-vrf-name argument. If the optional vrf keyword and receiver-vrf-name argument are not specified, the group-based VRF selection policy applies to RPF lookups originating in the global table.
Step 10	**end** **Example:** `Router(config-vrf)# end`	Closes the configuration session.

Example

This is sample example for configuring MLDP MVPN for configuring extranet using RSC:

Configuration on Source PE:

```
Router# enable
Router# conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# ip vrf blue1
Router(config-if)# rd 10:3
Router(config-if)# vpn id 10:3
```

```
Router(config-vrf)# route-target import 10:3
Router(config-vrf)# route-target export 10:3
Router(config-vrf)# mdt default mpls MLDP 2.2.2.2
mdt default mpls MLDP root-node
Router(config-if)# end
Router(config)# ip mroute vrf red 40.0.0.0 255.255.255.0 fallback-lookup vrf blue
Router(config-if)# end
```

Configuration on Receiver PE:

```
Router# enable
Router# conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# ip vrf blue1
Router(config-if)# rd 10:3
Router(config-if)# vpn id 10:3
Router(config-vrf)# route-target import 10:3
Router(config-vrf)# route-target export 10:3
Router(config-vrf)# mdt default mpls MLDP 2.2.2.2
Router(config)# ip mroute vrf red 40.0.0.0 255.255.255.0 fallback-lookup vrf blue
Router(config-if)# end
```

Configuring MLDP TE-FRR Support

TE-FRR provides link protection, however TE-FRR on MLDP provides link protection only for the single hop primary path. Node protection is not supported.These are the highlights:

- Backup tunnel support
- Backup bandwidth protection

For more information on MPLS TE-FRR, see MPLS Point-to-Multipoint Traffic Engineering.

Summary Steps

1. **enable**
2. **configure terminal**
3. **ip multicast mpls traffic-eng** [**range** {*access-list-number* | *access-list-name*}]
4. **mpls MLDP path traffic-eng**
5. **end**

DETAILED STEPS

	Command	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password when prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.

	Command	Purpose
Step 3	**ip multicast mpls traffic-eng** [**range** {access-list-number | access-list-name}] **Example:** Router(config)# ip multicast mpls traffic-eng	Enables IP multicast traffic on a tail end router enabled with MPLS TE P2MP functionality.
Step 4	**mpls MLDP path traffic-eng** **Example:** Router(config)# mpls MLDP path traffic-en	Configures MLDP to use traffic-eng tunnels.
Step 5	**end** **Example:** Router(config)# end	Closes the configuration session.

For more information, see MPLS Traffic Engineering (TE) - Fast Reroute (FRR) Link and Node Protection.

Configuring MLDP with PIM-based MVPN

MLDP with PIM-based MVPN supports MLDP coexistence with a PIM-based MVPN deployment. Using this feature, you can gradually introduce MLDP in an existing PIM-based MVPN environment, facilitating phased migration towards a complete LSM-based MVPN network infrastructure. If both the MLDP-based MVPN and GRE-based MVPN are configured, MDT selects PIM based MVPN by default. Configure the precedence for MLDP MVPN and PIM based MVPN using the **mdt preference** *option1 option2* command. This example sets MLDP MVPN precedence over PIM based MVPN:

```
Router(config-vrf)# mdt preference MLDP pim
```

MLDP Support with Load Balancing

MLDP supports load balancing of multicast traffic with Equal Cost Multipath (ECMP) links. For Load balancing to work with MLDP, use the **disable mpls MLDP forwarding recursive** command, which is enabled by default. Also, ensure that the **mpls MLDP path multipath** command is enabled for load balancing to function as expected.

Root Node Redundancy

Configure multiple root nodes in the network using the **mdt default mpls MLDP** *ip_address* command. The control plane builds a corresponding tree with root at the configured node to enable efficient forwarding. A node in the network selects the nearest root for optimal bandwidth usage. Also, in case a root node is unreachable (due to link failure, or router crash), the node switches to the next available root.

This example describes the root node redundancy configuration:

```
Router(config)# ip vrf blue1
Router(config-if)# rd 10:3
Router(config-if)# vpn id 10:3
Router(config-vrf)# route-target import 10:3
```

```
Router(config-vrf)# route-target export 10:3
Router(config-vrf)# mdt default mpls MLDP 2.2.2.2
Router(config-vrf)# mdt default mpls MLDP 5.5.5.5
```

Verification

Use these commands to verify the LSM-MLDP-based MVPN support configuration.

- To check the MLDP neighbors, use the **show mpls MLDP neighbors** command:

```
Router# show mpls MLDP neighbors
MLDP peer ID    : 3.3.3.3:0, uptime 00:41:41 Up,
  Target Adj     : Yes
  Session hndl   : 2
  Upstream count : 2
  Branch count   : 0
  Path count     : 1
  Path(s)        : 3.3.3.3           No LDP Tunnel20
  Nhop count     : 1
  Nhop list      : 3.3.3.3

 MLDP peer ID    : 2.2.2.2:0, uptime 00:17:42 Up,
  Target Adj     : No
  Session hndl   : 4
  Upstream count : 0
  Branch count   : 0
  Path count     : 1
  Path(s)        : 3.3.3.3           No LDP Tunnel20
  Nhop count     : 0
```

- To check the PIM neighbors, use the **show ip pim vrf** *vrf_name* **neighbor** command:

```
Router# show ip pim vrf blue neighbor
PIM Neighbor Table
Mode: B - Bidir Capable, DR - Designated Router, N - Default DR Priority,
      P - Proxy Capable, S - State Refresh Capable, G - GenID Capable
Neighbor          Interface                Uptime/Expires    Ver   DR
Address                                                            Prio/Mode
3.3.3.3           Lspvif1                  00:06:21/00:01:17 v2    1 / DR S P G
```

- To check the multicast routes for a given VRF, use **show ip mroute vrf** *vrf_name* **verbose** command:

```
Router# show ip mroute vrf blue verbose
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
       L - Local, P - Pruned, R - RP-bit set, F - Register flag,
       T - SPT-bit set, J - Join SPT, M - MSDP created entry, E - Extranet,
       X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
       U - URD, I - Received Source Specific Host Report,
       Z - Multicast Tunnel, z - MDT-data group sender,
       Y - Joined MDT-data group, y - Sending to MDT-data group,
       V - RD & Vector, v - Vector
Outgoing interface flags: H - Hardware switched, A - Assert winner
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode

(40.0.0.2, 232.0.1.4), 00:00:16/00:03:13, flags: sT
  Incoming interface: GigabitEthernet3/2/1, RPF nbr 0.0.0.0
  Outgoing interface list:
    Lspvif1, LSM MDT: B0000004 (default), Forward/Sparse, 00:00:16/00:03:13

(*, 224.0.1.40), 00:47:09/00:02:56, RP 0.0.0.0, flags: DPL
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list: Null
```

- To check the packet counters, use **show ip mroute vrf** *vrf_name* **count** command:

```
Router# show ip mroute vrf blue count
IP Multicast Statistics
2 routes using 1208 bytes of memory
2 groups, 0.50 average sources per group
Forwarding Counts: Pkt Count/Pkts per second/Avg Pkt Size/Kilobits per second
Other counts: Total/RPF failed/Other drops(OIF-null, rate-limit etc)

Group: 232.0.1.4, Source count: 1, Packets forwarded: 1333, Packets received: 1334
  Source: 40.0.0.2/32, Forwarding: 1333/20/46/7, Other: 1334/0/1

Group: 224.0.1.40, Source count: 0, Packets forwarded: 0, Packets received: 0
```

- To check the MFIB output and whether hardware switching or software switching is enabled, use **show ip mfib vrf** *vrf_name group_address* **verbose** command:

```
Router# show ip mfib vrf blue 232.0.1.4 verbose
Entry Flags:    C - Directly Connected, S - Signal, IA - Inherit A flag,
                ET - Data Rate Exceeds Threshold, K - Keepalive
                DDE - Data Driven Event, HW - Hardware Installed
I/O Item Flags: IC - Internal Copy, NP - Not platform switched,
                NS - Negate Signalling, SP - Signal Present,
                A - Accept, F - Forward, RA - MRIB Accept, RF - MRIB Forward,
                MA - MFIB Accept
Platform per slot HW-Forwarding Counts: Pkt Count/Byte Count
Platform Entry flags: HF - Hardware Forwarding, NP - Not platform switched,
                      PF - Partial Hardware Forwarding
Platform Interface flags: HW - Hardware Switched, NP - Not platform switched
Forwarding Counts: Pkt Count/Pkts per second/Avg Pkt Size/Kbits per second
Other counts:      Total/RPF failed/Other drops
I/O Item Counts:   FS Pkt Count/PS Pkt Count
VRF blue
 (40.0.0.2,232.0.1.4) Flags: K HW
   Platform Flags:  HW
   Slot 6: HW Forwarding: 912/41952, Platform Flags:  HF
   SW Forwarding: 0/0/0/0, Other: 1/0/1
   HW Forwarding:   912/20/46/7, Other: 0/0/0
   GigabitEthernet3/2/1 Flags: RA A MA
     Platform Flags:
   Lspvif1, LSM/B0000004 Flags: RF F NS
     Platform Flags:  HW
     CEF: Mid chain adjacency
     Pkts: 0/0
```

- To check the labels, use **show mpls forwarding-table** command:

```
Router# show mpls forwarding-table
Local Outgoing Prefix Bytes Label Outgoing Next Hop
Label      Label      or Tunnel Id     Switched      interface
16 Pop Label IPv4 VRF[V] 0 aggregate/blue
17 Pop Label  IPv4 VRF[V] 0 aggregate/red
18 [T] Pop Label  3.3.3.3/32 0 Tu20 point2point
19 [T]  25 2.2.2.2/32 0 Tu20 point2point
20 [T]  Pop Label 19.0.0.0/24 0 Tu20 point2point
22 [T]  No Label [mdt 55:1111 0][V] \9422 aggregate/red
23 [T]  No Label [mdt 55:2222 0][V] \9708          aggregate/blue
[T]     Forwarding through a LSP tunnel.
        View additional labelling info with the 'detail' option
```

- To display all the Replicate Output Chain Element (Replicate OCE) on the Forwarding Manager (FMAN) RP, use **show platform software mpls rp** *act-status* **replicate** command.

```
Router#show platform software mpls rp active replicate

Replicate-oce-list: 0x400000d2 (1 OCEs)
  OM: 0x42269b64
Replicate-oce-list: 0x400000d3 (1 OCEs)
  OM: 0x43ba2aec
Replicate-oce-list: 0x400000d4 (0 OCEs)
  OM: 0x422659bc
Replicate-oce-list: 0x400000d5 (0 OCEs)
  OM: 0x422658ac
```

- To display the Replicate OCE with the specified index value on FMAN RP, use **show platform software mpls rp** *act-status* **replicate index** *index-value* command.

Note You should run "**show platform software mpls rp active replicate**" first to see the all the replicated OCE on the FMAN RP.

```
Router#show platform software mpls fp active replicate

Replicate-oce-list: 0x84 (1 OCEs)
  AOM obj: 478, HW list: 0x11b19610 (created)

Router#show platform software mpls rp active replicate index 0x84 Replicate-oce-list
entries

 OCE             Type             Misc Info
--------------------------------------------------------------------------------
 0xa3            OBJ_LABEL        aom id: 494, HW info: 0x11b19e40
(created)
```

- To display all the replicated OCE on the FMAN FP, use **show platform software mpls fp** *act-status* **replicate** command.

```
Router#show platform software mpls fp active replicate

Replicate-oce-list: 0x400000d2 (1 OCEs)
  AOM obj: 352887, HW list: 0x11a65628 (created)
Replicate-oce-list: 0x400000d3 (1 OCEs)
  AOM obj: 352889, HW list: 0x10d4a518 (created)
Replicate-oce-list: 0x400000d4 (0 OCEs)
  AOM obj: 352891, HW list: 0x139e3d90 (created)
Replicate-oce-list: 0x400000d5 (0 OCEs)
  AOM obj: 352894, HW list: 0x139e7cb8 (created)
```

- To display the complete OCE chain used for forwarding traffic to a particular IPv4 multicast address, use **show platform hardware qfp active feature multicast v4mcast** *ip-address-mgroup ip-address-source* **vrf** *vrf-id* **extension** command.

```
Router#show platform hardware qfp active feature multicast v4mcast 239.1.1.1/32 vrf 2
extension

Root: 0x1187fc58
Flags: 0x000002
First leaf: 0x11887fa8
Number of nodes: 1
Number of leaves: 3
RPF i/f: 0x01fff7
Punt limit counter: 200
NS DCS Punt limit: 0x000001
RPF Fast Convergence Flags: 00000000
Secondary RPF interface: 00000000
```

```
RPF Fast Convergence Timer: 0
Extended leaf address: 0x89f80060

Node: 0x1187fc58
Cumulative Free Space: : 4
Cumulative Weight: : 3
Number of Children: : 3
Hw Addr: : 0x8b969440
Node Flags: : 0x000004
Software Child Ptr: : 0x1187fce0, 0x1187fd60, 0x11887fa8, 00000000
00000000, 00000000, 00000000
Hardware Child Ptr: : 0x89f8e440, 0x89f8e450, 0x89f8e460, 00000000
00000000, 00000000, 00000000

OCE Flags: : 0x000009
SW OCE chain ptr: 0x11884b48
HW OCE chain ptr: 0x895d59a0

OCE Type: Adjacency, Number of children: 1
Adj Type: : IPV4 Adjacency
Encap Len: : 0
L3 MTU: : 9216

Adj Flags: : 64
Fixup Flags: : 0
Interface Name: Lspvif0
Next Hop Address: : 00000000 00000000 00000000 00000000
Lisp locator status: : 00000000
Next HW OCE Ptr: : 0x895d5940

OCE Type: REPLICATE OCE, Number of children: 1
Replica_node: : 0x89fab440
Next HW OCE Ptr: : 0x895d5ab0

OCE Type: Label OCE, Number of children: 1
Label flags: : 0
Num Labels: : 1
Num Bk Labels: : 1
Out Labels: : 17
Out Backup Labels: : 0
Next HW OCE Ptr: : 0x895d5a70

OCE Type: Label OCE, Number of children: 1
Label flags: : 65
Num Labels: : 1
Num Bk Labels: : 0
Out Labels: : 3
Next HW OCE Ptr: : 0x895d59f0

OCE Type: Adjacency, Number of children: 0
Adj Type: : MPLS Adjacency
Encap Len: : 14
L3 MTU: : 1500
Adj Flags: : 0
Fixup Flags: : 0
Interface Name: GigabitEthernet0/1/0
Encap: : 00 24 14 f4 9d 00 00 21 d8 d4 a5 10 88 47
Next Hop Address: : 0b000002 00000000 00000000 00000000
Next HW OCE Ptr: : 00000000

OCE Flags: : 0x000002
SW OCE chain ptr: 0x118830d0
HW OCE chain ptr: 0x895d58f0
```

```
OCE Type: Adjacency, Number of children: 0
Adj Type: : IPV4 Adjacency
Encap Len: : 20
L3 MTU: : 1480
Adj Flags: : 0
Fixup Flags: : 2
Interface Name: Tunnel1
Encap: : 45 00 00 00 00 00 00 00 ff 67 39 94 c0 00 01 01
c0 00 01 01
Next Hop Address: : 00000000 00000000 00000000 00000000
Lisp locator status: : 00000000
Next HW OCE Ptr: : 00000000

OCE Flags: : 0x000009
SW OCE chain ptr: 0x1186c250
HW OCE chain ptr: 0x895d5650

OCE Type: Adjacency, Number of children: 0
Adj Type: : IPV4 Adjacency
Encap Len: : 14
L3 MTU: : 1500
Adj Flags: : 0
Fixup Flags: : 64
Interface Name: GigabitEthernet0/1/2
Encap: : 01 00 5e 00 00 00 00 21 d8 d4 a5 12 08 00
Next Hop Address: : e1000000 00000000 00000000 00000000
Lisp locator status: : 00000000
Next HW OCE Ptr: : 00000000

OCE Flags: : 0x000009
SW OCE chain ptr: 0x1186d478
HW OCE chain ptr: 0x895d5660

OCE Type: Adjacency, Number of children: 0
Adj Type: : IPV4 Adjacency
Encap Len: : 14
L3 MTU: : 1500

Adj Flags: : 0
Fixup Flags: : 64
Interface Name: GigabitEthernet0/1/4
Encap: : 01 00 5e 00 00 00 00 21 d8 d4 a5 14 08 00
Next Hop Address: : e1000000 00000000 00000000 00000000
Lisp locator status: : 00000000
Next HW OCE Ptr: : 00000000
```

- To display the complete OCE chain used for forwarding traffic to a particular IPv6 multicast address, use **show platform hardware qfp active feature multicast v6mcast** *ip-address-mgroup ip-address-source* **vrf** *vrf-id* **extension** command.

```
Router#show platform hardware qfp active feature multicast v6mcast FF04::10/128 vrf
503316482 extension

Root: 0x11b6c700
Flags: 0x000002
First leaf: 0x11e55bc8
Number of nodes: 1
Number of leaves: 3
RPF i/f: 0x01fff3
Punt limit counter: 200
NS DCS Punt limit: 0x000001
RPF Fast Convergence Flags: 00000000
Secondary RPF interface: 00000000
RPF Fast Convergence Timer: 0
```

```
Extended leaf address: 0x8ba18c90

Node: 0x11b6c700
Cumulative Free Space: : 4
Cumulative Weight: : 3
Number of Children: : 3
Hw Addr: : 0x8ba06c60
Node Flags: : 0x000004
Software Child Ptr: : 0x11b6dcb0, 0x11b6e0b0, 0x11e55bc8, 00000000
00000000, 00000000, 00000000
Hardware Child Ptr: : 0x8ba24060, 0x8ba24070, 0x8ba245f0, 00000000
00000000, 00000000, 00000000

OCE Flags: : 0x000009
SW OCE chain ptr: 0x11b71af0
HW OCE chain ptr: 0x895ffa40

OCE Type: Adjacency, Number of children: 1
Adj Type: : IPV6 Adjacency
Encap Len: : 0
L3 MTU: : 9216
Adj Flags: : 64
Fixup Flags: : 0
Interface Name: Lspvif0
Next Hop Address: : 00000000 00000000 00000000 00000000
Lisp locator status: : 00000000
Next HW OCE Ptr: : 0x895ffa20

OCE Type: Label OCE, Number of children: 1
Label flags: : 0
Num Labels: : 1
Num Bk Labels: : 1
Out Labels: : 2
Out Backup Labels: : 2
Next HW OCE Ptr: : 0x895ff9f0

OCE Type: Adjacency, Number of children: 1
Adj Type: : MPLS Adjacency
Encap Len: : 0
L3 MTU: : 9216
Adj Flags: : 64
Fixup Flags: : 0
Interface Name: Lspvif0
Next Hop Address: : 00000000 00000000 00000000 00000000
Next HW OCE Ptr: : 0x895ff980

OCE Type: REPLICATE OCE, Number of children: 1
Replica_node: : 0x8ba51060
Next HW OCE Ptr: : 0x895ffa60

OCE Type: Label OCE, Number of children: 1
Label flags: : 0
Num Labels: : 1
Num Bk Labels: : 1
Out Labels: : 17
Out Backup Labels: : 0
Next HW OCE Ptr: : 0x895ff7b0

OCE Type: Adjacency, Number of children: 0
Adj Type: : MPLS Adjacency
Encap Len: : 14
L3 MTU: : 1500
Adj Flags: : 0
Fixup Flags: : 0
```

```
Interface Name: GigabitEthernet0/1/0
Encap: : 00 24 14 f4 9d 00 00 21 d8 d4 a5 10 88 47
Next Hop Address: : 0b000002 00000000 00000000 00000000
Next HW OCE Ptr: : 00000000

OCE Flags: : 0x000009
SW OCE chain ptr: 0x11b6b800
HW OCE chain ptr: 0x895ff6a0

OCE Type: Adjacency, Number of children: 0
Adj Type: : IPV6 Adjacency
Encap Len: : 14
L3 MTU: : 1500
Adj Flags: : 0
Fixup Flags: : 64
Interface Name: GigabitEthernet0/1/2
Encap: : 33 33 00 00 00 00 21 d8 d4 a5 12 86 dd
Next Hop Address: : ff0e0000 00000000 00000000 00000000
Lisp locator status: : 00000000
Next HW OCE Ptr: : 00000000

OCE Flags: : 0x000009
SW OCE chain ptr: 0x11b6ba08
HW OCE chain ptr: 0x895ff6e0

OCE Type: Adjacency, Number of children: 0
Adj Type: : IPV6 Adjacency
Encap Len: : 14
L3 MTU: : 1500
Adj Flags: : 0
Fixup Flags: : 64
Interface Name: GigabitEthernet0/1/4
Encap: : 33 33 00 00 00 00 21 d8 d4 a5 14 86 dd
Next Hop Address: : ff0e0000 00000000 00000000 00000000
Lisp locator status: : 00000000
Next HW OCE Ptr: : 00000000

OCE Flags: : 0x00000a
SW OCE chain ptr: 0x11b6de20
HW OCE chain ptr: 0x895ff770

OCE Type: Adjacency, Number of children: 0
Adj Type: : IPV6 Adjacency
Encap Len: : 4
L3 MTU: : 1460
Adj Flags: : 2
Fixup Flags: : 2
Interface Name: Tunnel5
Encap: : f8 00 01 47
Next Hop Address: : 00000000 00000000 00000000 00000000
Lisp locator status: : 00000000
Next HW OCE Ptr: : 00000000

Root: 0x11e4f428
Flags: 00000000
First leaf: 0x11e51b90
Number of nodes: 1
Number of leaves: 3
RPF i/f: 0x0003fd
Punt limit counter: 200
NS DCS Punt limit: 0x000001
RPF Fast Convergence Flags: 00000000
Secondary RPF interface: 00000000
RPF Fast Convergence Timer: 0
```

```
Extended leaf address: 0x8ba21210

Node: 0x11e4f428
Cumulative Free Space: : 4
Cumulative Weight: : 3
Number of Children: : 3
Hw Addr: : 0x8ba0c560
Node Flags: : 0x000004
Software Child Ptr: : 0x11e424b8, 0x11e332b8, 0x11e51b90, 00000000

Root: 0x11e50f20
Flags: 00000000
First leaf: 0x11e51b90
Number of nodes: 1
Number of leaves: 3
RPF i/f: 0x0003fd
Punt limit counter: 200
NS DCS Punt limit: 0x000001
RPF Fast Convergence Flags: 00000000
Secondary RPF interface: 00000000
RPF Fast Convergence Timer: 0
Extended leaf address: 0x8ba212a0

Node: 0x11e50f20
Cumulative Free Space: : 4
Cumulative Weight: : 3
Number of Children: : 3
Hw Addr: : 0x8ba0c560
Node Flags: : 0x000004
Software Child Ptr: : 0x11e424b8, 0x11e56f98, 0x11e51b90, 00000000
00000000, 00000000, 00000000
Hardware Child Ptr: : 0x8ba247a0, 0x8ba24750, 0x8ba24740, 00000000
00000000, 00000000, 00000000

OCE Flags: : 0x000009
SW OCE chain ptr: 0x11b6ba08
HW OCE chain ptr: 0x895ff6e0

OCE Type: Adjacency, Number of children: 0
Adj Type: : IPV6 Adjacency
Encap Len: : 14
L3 MTU: : 1500
Adj Flags: : 0
Fixup Flags: : 64
Interface Name: GigabitEthernet0/1/4
Encap: : 33 33 00 00 00 00 00 21 d8 d4 a5 14 86 dd
Next Hop Address: : ff0e0000 00000000 00000000 00000000
Lisp locator status: : 00000000
Next HW OCE Ptr: : 00000000

OCE Flags: : 0x000009
SW OCE chain ptr: 0x11b71af0
HW OCE chain ptr: 0x895ffa40

OCE Type: Adjacency, Number of children: 1
Adj Type: : IPV6 Adjacency
Encap Len: : 0
L3 MTU: : 9216
Adj Flags: : 64
Fixup Flags: : 0
Interface Name: Lspvif0
Next Hop Address: : 00000000 00000000 00000000 00000000
Lisp locator status: : 00000000
Next HW OCE Ptr: : 0x895ffa20
```

```
OCE Type: Label OCE, Number of children: 1
Label flags: : 0
Num Labels: : 1
Num Bk Labels: : 1
Out Labels: : 2
Out Backup Labels: : 2
Next HW OCE Ptr: : 0x895ff9f0

OCE Type: Adjacency, Number of children: 1
Adj Type: : MPLS Adjacency
Encap Len: : 0
L3 MTU: : 9216
Adj Flags: : 64
Fixup Flags: : 0
Interface Name: Lspvif0
Next Hop Address: : 00000000 00000000 00000000 00000000
Next HW OCE Ptr: : 0x895ff980

OCE Type: REPLICATE OCE, Number of children: 1
Replica_node: : 0x8ba51060
Next HW OCE Ptr: : 0x895ffa60

OCE Type: Label OCE, Number of children: 1
Label flags: : 0
Num Labels: : 1
Num Bk Labels: : 1
Out Labels: : 17
Out Backup Labels: : 0
Next HW OCE Ptr: : 0x895ff7b0

OCE Type: Adjacency, Number of children: 0
Adj Type: : MPLS Adjacency
Encap Len: : 14
L3 MTU: : 1500
Adj Flags: : 0
Fixup Flags: : 0
Interface Name: GigabitEthernet0/1/0
Encap: : 00 24 14 f4 9d 00 00 21 d8 d4 a5 10 88 47

Next Hop Address: : 0b000002 00000000 00000000 00000000
Next HW OCE Ptr: : 00000000

OCE Flags: : 0x000003
SW OCE chain ptr: 0x11b6b800
HW OCE chain ptr: 0x895ff6a0

OCE Type: Adjacency, Number of children: 0
Adj Type: : IPV6 Adjacency
Encap Len: : 14
L3 MTU: : 1500
Adj Flags: : 0
Fixup Flags: : 64
Interface Name: GigabitEthernet0/1/2
Encap: : 33 33 00 00 00 00 00 21 d8 d4 a5 12 86 dd
Next Hop Address: : ff0e0000 00000000 00000000 00000000
Lisp locator status: : 00000000
Next HW OCE Ptr: : 00000000
```

- To display the complete OCE chain used for handling incoming MPLS packets with the particular label, use **show platform hardware qfp active feature cef-mpls prefix mpls** *mpls-lable* **exact** command.

```
Router# show platform hardware qfp active feature cef-mpls prefix mpls 17 exact
```

```
Gtrie Node Type: Leaf Node
HW Content: : 0a000000 00000f00 00000000 8bb08a30
QPPB QoS Precedence valid: 0
QoS Precedence: 0
QPPB QoS Group valid: 0
QoS Group: 0
BGPPA Traffic Index valid: 0
BGPPA Traffic Index: 0
TBLF refcount: 2
TBLF application lf handle: 0
CTS src_sgt: 0
CTS dst_sgt: 0
Prefix Length: 20
Prefix: 00 0d 00
Lisp local eid: 0
Lisp remote eid: 0
Lisp locator status bits: 0
Lisp dynamic configured eid: 0
Lisp dynamic discovered eid: 0

OCE Type: EOS OCE, Number of children: 2
Next HW OCE Ptr: : 0x8bb07e10, 0x8bb07e00

OCE Type: REPLICATE OCE, Number of children: 2
Replica_node: : 0x8ca90a20
Next HW OCE Ptr: : 0x8bb07eb0, 0x8bb08840

OCE Type: Label OCE, Number of children: 1
Label flags: : 64
Num Labels: : 1
Num Bk Labels: : 0
Out Labels: : 1048577
Next HW OCE Ptr: : 0x8bb07e60

OCE Type: Interface OCE, Number of children: 1
Next HW OCE Ptr: : 0x8bb07e40
Interface Name: Lspvif20

OCE Type: Lookup OCE, Number of children: 0
Lookup flags: : 1
Table Type: : 0
Lookup table ID: : 0

OCE Type: Label OCE, Number of children: 1
Label flags: : 0
Num Labels: : 1
Num Bk Labels: : 1
Out Labels: : 88
Out Backup Labels: : 0
Next HW OCE Ptr: : 0x8bb06ca0

OCE Type: Adjacency, Number of children: 0
Adj Type: : MPLS Adjacency
Encap Len: : 14
L3 MTU: : 1500

Adj Flags: : 0
Fixup Flags: : 0
Interface Name: GigabitEthernet0/1/0
Encap: : 00 0e 39 88 70 19 00 21 d8 60 c0 10 88 47
Next Hop Address: : 0f000001 00000000 00000000 00000000
Next HW OCE Ptr: : 00000000
```

```
OCE Type: REPLICATE OCE, Number of children: 2
Replica_node: : 0x8ca90a00
Next HW OCE Ptr: : 0x8bb07e70, 0x8bb08840

OCE Type: Label OCE, Number of children: 1
Label flags: : 64
Num Labels: : 1
Num Bk Labels: : 0
Out Labels: : 1048577
Next HW OCE Ptr: : 0x8bb07e50

OCE Type: Interface OCE, Number of children: 1
Next HW OCE Ptr: : 0x8bb001f0
Interface Name: Lspvif20

OCE Type: Lookup OCE, Number of children: 0
Lookup flags: : 0
Table Type: : 1
Lookup table ID: : 2

OCE Type: Label OCE, Number of children: 1
Label flags: : 0
Num Labels: : 1
Num Bk Labels: : 1
Out Labels: : 88
Out Backup Labels: : 0
Next HW OCE Ptr: : 0x8bb06ca0

OCE Type: Adjacency, Number of children: 0
Adj Type: : MPLS Adjacency
Encap Len: : 14

L3 MTU: : 1500
Adj Flags: : 0
Fixup Flags: : 0
Interface Name: GigabitEthernet0/1/0
Encap: : 00 0e 39 88 70 19 00 21 d8 60 c0 10 88 47
Next Hop Address: : 0f000001 00000000 00000000 00000000
Next HW OCE Ptr: : 00000000
```

Sample Configuration for MLDP MVPN

You can configure MLDP MVPN in these two modes:

- Source Specific Mode (SSM)
- Sparse Mode (SM)

Configuration Example Using SSM Mode

These examples are based on the topology shown in Figure 18-1. Consider these scenarios while configuring MLDP MVPN using SSM mode:

- MLDP MVPN Extranet SSC
- MLDP MVPN Extranet RSC
- MLDP MVPN Intranet

MLDP MVPN Extranet SSC

Configuration on PE1 Router (Source PE):

```
ip vrf red2
 rd 10:2
 vpn id 10:2
 mdt default mpls MLDP 4.4.4.4
 mdt data mpls MLDP 100
 mdt data threshold 20
 route-target export 10:2
 route-target import 10:2
!
ip vrf red3
 rd 10:3
 vpn id 10:3
 mdt default mpls MLDP 4.4.4.4
 mdt data mpls MLDP 100
 mdt data threshold 20
 route-target export 10:3
 route-target import 10:3
!
ip multicast-routing
ip multicast-routing vrf red2
ip multicast-routing vrf red3

interface Loopback1
 ip address 1.1.1.1 255.255.255.255
 ip pim sparse-mode
!
interface Loopback102
 ip vrf forwarding red2
 ip address 101.2.0.2 255.255.255.255
 ip pim sparse-mode
!
interface Loopback103
 ip vrf forwarding red3
 ip address 101.3.0.2 255.255.255.255
 ip pim sparse-mode

interface GigabitEthernet1/22.2
 encapsulation dot1Q 2
 ip vrf forwarding red2
 ip address 12.2.0.1 255.255.0.0
 ip pim sparse-mode
!
interface TenGigabitEthernet8/1
 ip address 10.1.1.1 255.255.255.0
 ip ospf 1 area 0
 load-interval 30
 mpls ip
 mpls label protocol ldp

router ospf 1
 router-id 1.1.1.1
 network 1.1.1.1 0.0.0.0 area 0
!
router bgp 100
 bgp log-neighbor-changes
 neighbor 2.2.2.2 remote-as 100
 neighbor 2.2.2.2 update-source Loopback1
 neighbor 3.3.3.3 remote-as 100
 neighbor 3.3.3.3 update-source Loopback1
 neighbor 4.4.4.4 remote-as 100
```

```
 neighbor 4.4.4.4 update-source Loopback1
 !
 address-family ipv4
  neighbor 2.2.2.2 activate
  neighbor 3.3.3.3 activate
  neighbor 4.4.4.4 activate
  no auto-summary
 exit-address-family
 !
 address-family vpnv4
  neighbor 2.2.2.2 activate
  neighbor 2.2.2.2 send-community both
  neighbor 3.3.3.3 activate
  neighbor 3.3.3.3 send-community both
 exit-address-family
 !
 address-family ipv4 mdt
  neighbor 2.2.2.2 activate
  neighbor 2.2.2.2 send-community both
  neighbor 3.3.3.3 activate
  neighbor 3.3.3.3 send-community both
 exit-address-family
 !
 address-family ipv4 vrf red2
  redistribute static
  redistribute connected
  neighbor 2.2.2.2 remote-as 100
  neighbor 2.2.2.2 activate
  neighbor 2.2.2.2 send-community both
  neighbor 3.3.3.3 remote-as 100
  neighbor 3.3.3.3 activate
  neighbor 3.3.3.3 send-community both
 exit-address-family
 !
 address-family ipv4 vrf red3
  redistribute static
  redistribute connected
  neighbor 2.2.2.2 remote-as 100
  neighbor 2.2.2.2 activate
  neighbor 2.2.2.2 send-community both
  neighbor 3.3.3.3 remote-as 100
  neighbor 3.3.3.3 activate
  neighbor 3.3.3.3 send-community both
 exit-address-family

ip pim vrf red2 ssm default
ip pim vrf red3 ssm default
ip mroute vrf red3 12.2.0.0 255.255.0.0 fallback-lookup vrf red2
```

Configuration on PE Router:

```
interface Loopback1
 ip address 4.4.4.4 255.255.255.255

interface GigabitEthernet2/10
 ip address 20.1.1.2 255.255.255.0
 ip ospf 1 area 0
 load-interval 30
 mpls ip
 mpls label protocol ldp

interface GigabitEthernet2/20
 ip address 30.1.1.2 255.255.255.0
```

```
 ip ospf 1 area 0
 mpls ip
 mpls label protocol ldp

interface TenGigabitEthernet4/0/0
 ip address 10.1.1.2 255.255.255.0
 ip ospf 1 area 0
 load-interval 30
 mpls ip
 mpls label protocol ldp

router ospf 1
 router-id 4.4.4.4
 network 4.4.4.4 0.0.0.0 area 0
!
router bgp 100
 bgp log-neighbor-changes
 neighbor 1.1.1.1 remote-as 100
 neighbor 2.2.2.2 remote-as 100
 neighbor 3.3.3.3 remote-as 100
 !
 address-family ipv4
  neighbor 1.1.1.1 activate
  neighbor 2.2.2.2 activate
  neighbor 3.3.3.3 activate
  no auto-summary
 exit-address-family
```

Configuration on PE2 Router (Receiver PE):

```
ip vrf red3
 rd 10:3
 vpn id 10:3
 mdt default mpls MLDP 4.4.4.4
 mdt data mpls MLDP 100
 mdt data threshold 20
 route-target export 10:3
 route-target import 10:3
!
ip multicast-routing
ip multicast-routing vrf red3

interface Loopback1
 ip address 2.2.2.2 255.255.255.255
 ip pim sparse-mode
!
interface Loopback103
 ip vrf forwarding red3
 ip address 102.3.0.2 255.255.255.255
 ip pim sparse-mode
!
interface GigabitEthernet4/0/0
 ip address 20.1.1.1 255.255.255.0
 ip ospf 1 area 0
 load-interval 30
 negotiation auto
 mpls ip
 mpls label protocol ldp
!
interface GigabitEthernet4/0/1.3
 encapsulation dot1Q 3
 ip vrf forwarding red3
 ip address 22.2.0.1 255.255.0.0
```

```
 ip pim sparse-mode
!
router ospf 1
 router-id 2.2.2.2
 network 2.2.2.2 0.0.0.0 area 0
!
router bgp 100
 bgp log-neighbor-changes
 neighbor 1.1.1.1 remote-as 100
 neighbor 1.1.1.1 update-source Loopback1
 neighbor 3.3.3.3 remote-as 100
 neighbor 3.3.3.3 update-source Loopback1
 neighbor 4.4.4.4 remote-as 100
 neighbor 4.4.4.4 update-source Loopback1
 !
 address-family ipv4
  neighbor 1.1.1.1 activate
  neighbor 3.3.3.3 activate
  neighbor 4.4.4.4 activate
  no auto-summary
 exit-address-family
 !
 address-family vpnv4
  neighbor 1.1.1.1 activate
  neighbor 1.1.1.1 send-community both
  neighbor 3.3.3.3 activate
  neighbor 3.3.3.3 send-community both
 exit-address-family
 !
 address-family ipv4 mdt
  neighbor 1.1.1.1 activate
  neighbor 1.1.1.1 send-community both
  neighbor 3.3.3.3 activate
  neighbor 3.3.3.3 send-community both
 exit-address-family
 !
 address-family ipv4 vrf red3
  redistribute static
  redistribute connected
  neighbor 1.1.1.1 remote-as 100
  neighbor 1.1.1.1 activate
  neighbor 1.1.1.1 send-community both
  neighbor 3.3.3.3 remote-as 100
  neighbor 3.3.3.3 activate
  neighbor 3.3.3.3 send-community both
 exit-address-family
!
ip pim vrf red3 ssm default
ip mroute vrf red3 12.2.0.0 255.255.0.0 101.3.0.2
```

Configuraton on PE3 Router (Receiver PE)

```
ip vrf red3
 rd 10:3
 vpn id 10:3
 mdt default mpls MLDP 4.4.4.4
 mdt data mpls MLDP 100
 mdt data threshold 20
 route-target export 10:3
 route-target import 10:3
!
ip multicast-routing
```

```
ip multicast-routing vrf red3
!
interface Loopback1
 ip address 3.3.3.3 255.255.255.255
 ip pim sparse-mode
!
interface Loopback103
 ip vrf forwarding red3
 ip address 103.3.0.2 255.255.255.255
 ip pim sparse-mode
!
interface GigabitEthernet3/2/0.3
 encapsulation dot1Q 3
 ip vrf forwarding red3
 ip address 32.2.0.1 255.255.0.0
 ip pim sparse-mode
 ip igmp version 3
!
interface GigabitEthernet3/2/1
 ip address 30.1.1.1 255.255.255.0
 ip ospf 1 area 0
 load-interval 30
 negotiation auto
 mpls ip
 mpls label protocol ldp
!
router ospf 1
 router-id 3.3.3.3
 network 3.3.3.3 0.0.0.0 area 0
!
router bgp 100
 bgp log-neighbor-changes
 neighbor 1.1.1.1 remote-as 100
 neighbor 1.1.1.1 update-source Loopback1
 neighbor 2.2.2.2 remote-as 100
 neighbor 2.2.2.2 update-source Loopback1
 neighbor 4.4.4.4 remote-as 100
 neighbor 4.4.4.4 update-source Loopback1
 !
 address-family ipv4
  neighbor 1.1.1.1 activate
  neighbor 2.2.2.2 activate
  neighbor 4.4.4.4 activate
  no auto-summary
 exit-address-family
 !
 address-family vpnv4
  neighbor 1.1.1.1 activate
  neighbor 1.1.1.1 send-community both
  neighbor 2.2.2.2 activate
  neighbor 2.2.2.2 send-community both
 exit-address-family
 !
 address-family ipv4 mdt
  neighbor 1.1.1.1 activate
  neighbor 1.1.1.1 send-community both
  neighbor 2.2.2.2 activate
  neighbor 2.2.2.2 send-community both
 exit-address-family
 !
 address-family ipv4 vrf red3
  redistribute static
  redistribute connected
  neighbor 1.1.1.1 remote-as 100
```

```
  neighbor 1.1.1.1 activate
  neighbor 1.1.1.1 send-community both
  neighbor 2.2.2.2 remote-as 100
  neighbor 2.2.2.2 activate
  neighbor 2.2.2.2 send-community both
 exit-address-family
!
ip pim vrf red3 ssm default
ip mroute vrf red3 12.2.0.0 255.255.0.0 101.3.0.2
```

MLDP MVPN Extranet RSC

Configuration on PE1 Router (Source PE)

```
ip vrf red2
 rd 10:2
 vpn id 10:2
 mdt default mpls MLDP 4.4.4.4
 mdt data mpls MLDP 100
 mdt data threshold 20
 route-target export 10:2
 route-target import 10:2
!
ip multicast-routing
ip multicast-routing vrf red2
!
interface Loopback1
 ip address 1.1.1.1 255.255.255.255
 ip pim sparse-mode
!
interface Loopback102
 ip vrf forwarding red2
 ip address 101.2.0.2 255.255.255.255
 ip pim sparse-mode
!
interface GigabitEthernet1/22.2
 encapsulation dot1Q 2
 ip vrf forwarding red2
 ip address 12.2.0.1 255.255.0.0
 ip pim sparse-mode
!
interface TenGigabitEthernet8/1
 ip address 10.1.1.1 255.255.255.0
 ip ospf 1 area 0
 load-interval 30
 mpls ip
 mpls label protocol ldp
!
router ospf 1
 router-id 1.1.1.1
 network 1.1.1.1 0.0.0.0 area 0
!
router bgp 100
 bgp log-neighbor-changes
 neighbor 2.2.2.2 remote-as 100
 neighbor 2.2.2.2 update-source Loopback1
 neighbor 3.3.3.3 remote-as 100
 neighbor 3.3.3.3 update-source Loopback1
 neighbor 4.4.4.4 remote-as 100
 neighbor 4.4.4.4 update-source Loopback1
 !
 address-family ipv4
```

```
  neighbor 2.2.2.2 activate
  neighbor 3.3.3.3 activate
  neighbor 4.4.4.4 activate
  no auto-summary
 exit-address-family
 !
 address-family vpnv4
  neighbor 2.2.2.2 activate
  neighbor 2.2.2.2 send-community both
  neighbor 3.3.3.3 activate
  neighbor 3.3.3.3 send-community both
 exit-address-family
 !
 address-family ipv4 mdt
  neighbor 2.2.2.2 activate
  neighbor 2.2.2.2 send-community both
  neighbor 3.3.3.3 activate
  neighbor 3.3.3.3 send-community both
 exit-address-family
 !
 address-family ipv4 vrf red2
  redistribute static
  redistribute connected
  neighbor 2.2.2.2 remote-as 100
  neighbor 2.2.2.2 activate
  neighbor 2.2.2.2 send-community both
  neighbor 3.3.3.3 remote-as 100
  neighbor 3.3.3.3 activate
  neighbor 3.3.3.3 send-community both
 exit-address-family
!
ip pim vrf red2 ssm default
```

Configuration on P Router (Core Router)

```
interface Loopback1
 ip address 4.4.4.4 255.255.255.255
!
interface GigabitEthernet2/10
 ip address 20.1.1.2 255.255.255.0
 ip ospf 1 area 0
 load-interval 30
 mpls ip
 mpls label protocol ldp
!
interface GigabitEthernet2/20
 ip address 30.1.1.2 255.255.255.0
 ip ospf 1 area 0
 mpls ip
 mpls label protocol ldp
!
interface TenGigabitEthernet4/0/0
 ip address 10.1.1.2 255.255.255.0
 ip ospf 1 area 0
 load-interval 30
 mpls ip
 mpls label protocol ldp
 mls qos trust dscp
!
router ospf 1
 router-id 4.4.4.4
 network 4.4.4.4 0.0.0.0 area 0
```

```
!
router bgp 100
 bgp log-neighbor-changes
 neighbor 1.1.1.1 remote-as 100
 neighbor 2.2.2.2 remote-as 100
 neighbor 3.3.3.3 remote-as 100
 !
 address-family ipv4
  neighbor 1.1.1.1 activate
  neighbor 2.2.2.2 activate
  neighbor 3.3.3.3 activate
  no auto-summary
 exit-address-family
!
```

Configuration ond PE2 Router (Receiver PE)

```
ip vrf red2
 rd 10:2
 vpn id 10:2
 mdt default mpls MLDP 4.4.4.4
 mdt data mpls MLDP 100
 mdt data threshold 20
 route-target export 10:2
 route-target import 10:2
!
ip vrf red3
 rd 10:3
 vpn id 10:3
 mdt default mpls MLDP 4.4.4.4
 mdt data mpls MLDP 100
 mdt data threshold 20
 route-target export 10:3
 route-target import 10:3
!
ip multicast-routing
ip multicast-routing vrf red3
ip multicast-routing vrf red2
!
interface Loopback1
 ip address 2.2.2.2 255.255.255.255
 ip pim sparse-mode
!
interface Loopback102
 ip vrf forwarding red2
 ip address 102.2.0.2 255.255.255.255
 ip pim sparse-mode
!
interface Loopback103
 ip vrf forwarding red3
 ip address 102.3.0.2 255.255.255.255
 ip pim sparse-mode
!
interface GigabitEthernet4/0/0
 ip address 20.1.1.1 255.255.255.0
 ip ospf 1 area 0
 load-interval 30
 negotiation auto
 mpls ip
 mpls label protocol ldp
!
interface GigabitEthernet4/0/1.3
 encapsulation dot1Q 3
```

```
 ip vrf forwarding red3
 ip address 22.2.0.1 255.255.0.0
 ip pim sparse-mode
!
router ospf 1
 router-id 2.2.2.2
 network 2.2.2.2 0.0.0.0 area 0
!
router bgp 100
 bgp log-neighbor-changes
 neighbor 1.1.1.1 remote-as 100
 neighbor 1.1.1.1 update-source Loopback1
 neighbor 3.3.3.3 remote-as 100
 neighbor 3.3.3.3 update-source Loopback1
 neighbor 4.4.4.4 remote-as 100
 neighbor 4.4.4.4 update-source Loopback1
 !
 address-family ipv4
  neighbor 1.1.1.1 activate
  neighbor 3.3.3.3 activate
  neighbor 4.4.4.4 activate
  no auto-summary
 exit-address-family
 !
 address-family vpnv4
  neighbor 1.1.1.1 activate
  neighbor 1.1.1.1 send-community both
  neighbor 3.3.3.3 activate
  neighbor 3.3.3.3 send-community both
 exit-address-family
 !
 address-family ipv4 mdt
  neighbor 1.1.1.1 activate
  neighbor 1.1.1.1 send-community both
  neighbor 3.3.3.3 activate
  neighbor 3.3.3.3 send-community both
 exit-address-family
 !
 address-family ipv4 vrf red2
  redistribute static
  redistribute connected
  neighbor 1.1.1.1 remote-as 100
  neighbor 1.1.1.1 activate
  neighbor 1.1.1.1 send-community both
  neighbor 3.3.3.3 remote-as 100
  neighbor 3.3.3.3 activate
  neighbor 3.3.3.3 send-community both
 exit-address-family
 !
 address-family ipv4 vrf red3
  redistribute static
  redistribute connected
  neighbor 1.1.1.1 remote-as 100
  neighbor 1.1.1.1 activate
  neighbor 1.1.1.1 send-community both
  neighbor 3.3.3.3 remote-as 100
  neighbor 3.3.3.3 activate
  neighbor 3.3.3.3 send-community both
 exit-address-family
!
ip pim vrf red3 ssm default
ip pim vrf red2 ssm default
ip mroute vrf red3 12.2.0.0 255.255.0.0 fallback-lookup vrf red2
```

Configuration on PE3 Router (Receiver PE)

```
ip vrf red2
 rd 10:2
 vpn id 10:2
 mdt default mpls MLDP 4.4.4.4
 mdt data mpls MLDP 100
 mdt data threshold 20
 route-target export 10:2
 route-target import 10:2
!
ip vrf red3
 rd 10:3
 vpn id 10:3
 mdt default mpls MLDP 4.4.4.4
 mdt data mpls MLDP 100
 mdt data threshold 20
 route-target export 10:3
 route-target import 10:3
!
ip multicast-routing
ip multicast-routing vrf red3
ip multicast-routing vrf red2
!
interface Loopback1
 ip address 3.3.3.3 255.255.255.255
 ip pim sparse-mode
!
interface Loopback102
 ip vrf forwarding red2
 ip address 103.2.0.2 255.255.255.255
 ip pim sparse-mode
!
interface Loopback103
 ip vrf forwarding red3
 ip address 103.3.0.2 255.255.255.255
 ip pim sparse-mode
!
interface GigabitEthernet3/2/0.3
 encapsulation dot1Q 3
 ip vrf forwarding red3
 ip address 32.2.0.1 255.255.0.0
 ip pim sparse-mode
 ip igmp version 3
!
interface GigabitEthernet3/2/1
 ip address 30.1.1.1 255.255.255.0
 ip ospf 1 area 0
 load-interval 30
 negotiation auto
 mpls ip
 mpls label protocol ldp
!
router ospf 1
 router-id 3.3.3.3
 network 3.3.3.3 0.0.0.0 area 0
!
router bgp 100
 bgp log-neighbor-changes
 neighbor 1.1.1.1 remote-as 100
 neighbor 1.1.1.1 update-source Loopback1
 neighbor 2.2.2.2 remote-as 100
 neighbor 2.2.2.2 update-source Loopback1
 neighbor 4.4.4.4 remote-as 100
```

```
 neighbor 4.4.4.4 update-source Loopback1
 !
 address-family ipv4
  neighbor 1.1.1.1 activate
  neighbor 2.2.2.2 activate
  neighbor 4.4.4.4 activate
  no auto-summary
 exit-address-family
 !
 address-family vpnv4
  neighbor 1.1.1.1 activate
  neighbor 1.1.1.1 send-community both
  neighbor 2.2.2.2 activate
  neighbor 2.2.2.2 send-community both
 exit-address-family
 !
 address-family ipv4 mdt
  neighbor 1.1.1.1 activate
  neighbor 1.1.1.1 send-community both
  neighbor 2.2.2.2 activate
  neighbor 2.2.2.2 send-community both
 exit-address-family
 !
 address-family ipv4 vrf red2
  redistribute static
  redistribute connected
  neighbor 1.1.1.1 remote-as 100
  neighbor 1.1.1.1 activate
  neighbor 1.1.1.1 send-community both
  neighbor 2.2.2.2 remote-as 100
  neighbor 2.2.2.2 activate
  neighbor 2.2.2.2 send-community both
 exit-address-family
 !
 address-family ipv4 vrf red3
  redistribute static
  redistribute connected
  neighbor 1.1.1.1 remote-as 100
  neighbor 1.1.1.1 activate
  neighbor 1.1.1.1 send-community both
  neighbor 2.2.2.2 remote-as 100
  neighbor 2.2.2.2 activate
  neighbor 2.2.2.2 send-community both
 exit-address-family
!
ip pim vrf red3 ssm default
ip pim vrf red2 ssm default
ip mroute vrf red3 12.2.0.0 255.255.0.0 fallback-lookup vrf red2
```

MLDP MVPN Intranet

Configuration ond PE1 Router (Source PE)

```
ip vrf red2
 rd 10:2
 vpn id 10:2
 mdt default mpls MLDP 4.4.4.4
 mdt data mpls MLDP 100
 mdt data threshold 20
 route-target export 10:2
 route-target import 10:2
!
ip multicast-routing
```

```
ip multicast-routing vrf red2
!
interface Loopback1
 ip address 1.1.1.1 255.255.255.255
 ip pim sparse-mode
!
interface Loopback102
 ip vrf forwarding red2
 ip address 101.2.0.2 255.255.255.255
 ip pim sparse-mode
!
interface GigabitEthernet1/22.2
 encapsulation dot1Q 2
 ip vrf forwarding red2
 ip address 12.2.0.1 255.255.0.0
 ip pim sparse-mode
!
interface TenGigabitEthernet8/1
 ip address 10.1.1.1 255.255.255.0
 ip ospf 1 area 0
 load-interval 30
 mpls ip
 mpls label protocol ldp
!
router ospf 1
 router-id 1.1.1.1
 network 1.1.1.1 0.0.0.0 area 0
!
router bgp 100
 bgp log-neighbor-changes
 neighbor 2.2.2.2 remote-as 100
 neighbor 2.2.2.2 update-source Loopback1
 neighbor 3.3.3.3 remote-as 100
 neighbor 3.3.3.3 update-source Loopback1
 neighbor 4.4.4.4 remote-as 100
 neighbor 4.4.4.4 update-source Loopback1
 !
 address-family ipv4
  neighbor 2.2.2.2 activate
  neighbor 3.3.3.3 activate
  neighbor 4.4.4.4 activate
  no auto-summary
 exit-address-family
 !
 address-family vpnv4
  neighbor 2.2.2.2 activate
  neighbor 2.2.2.2 send-community both
  neighbor 3.3.3.3 activate
  neighbor 3.3.3.3 send-community both
 exit-address-family
 !
 address-family ipv4 mdt
  neighbor 2.2.2.2 activate
  neighbor 2.2.2.2 send-community both
  neighbor 3.3.3.3 activate
  neighbor 3.3.3.3 send-community both
 exit-address-family
 !
 address-family ipv4 vrf red2
  redistribute static
  redistribute connected
  neighbor 2.2.2.2 remote-as 100
  neighbor 2.2.2.2 activate
  neighbor 2.2.2.2 send-community both
```

```
  neighbor 3.3.3.3 remote-as 100
  neighbor 3.3.3.3 activate
  neighbor 3.3.3.3 send-community both
 exit-address-family
!
ip pim vrf red2 ssm default
```

Configuration on P Router (Core Router)

```
interface Loopback1
 ip address 4.4.4.4 255.255.255.255
!
interface GigabitEthernet2/10
 ip address 20.1.1.2 255.255.255.0
 ip ospf 1 area 0
 load-interval 30
 mpls ip
 mpls label protocol ldp
!
interface GigabitEthernet2/20
 ip address 30.1.1.2 255.255.255.0
 ip ospf 1 area 0
 mpls ip
 mpls label protocol ldp
!
interface TenGigabitEthernet4/0/0
 ip address 10.1.1.2 255.255.255.0
 ip ospf 1 area 0
 load-interval 30
 mpls ip
 mpls label protocol ldp
 mls qos trust dscp
!
router ospf 1
 router-id 4.4.4.4
 network 4.4.4.4 0.0.0.0 area 0
!
router bgp 100
 bgp log-neighbor-changes
 neighbor 1.1.1.1 remote-as 100
 neighbor 2.2.2.2 remote-as 100
 neighbor 3.3.3.3 remote-as 100
 !
 address-family ipv4
  neighbor 1.1.1.1 activate
  neighbor 2.2.2.2 activate
  neighbor 3.3.3.3 activate
  no auto-summary
 exit-address-family
!
```

Configuration on PE2 Router (Receiver PE)

```
ip vrf red2
 rd 10:2
 vpn id 10:2
 mdt default mpls MLDP 4.4.4.4
 mdt data mpls MLDP 100
 mdt data threshold 20
 route-target export 10:2
 route-target import 10:2
```

```
!
ip multicast-routing
ip multicast-routing vrf red2
!
interface Loopback1
 ip address 2.2.2.2 255.255.255.255
 ip pim sparse-mode
!
interface Loopback102
 ip vrf forwarding red2
 ip address 102.2.0.2 255.255.255.255
 ip pim sparse-mode
!
interface GigabitEthernet4/0/0
 ip address 20.1.1.1 255.255.255.0
 ip ospf 1 area 0
 load-interval 30
 negotiation auto
 mpls ip
 mpls label protocol ldp
!
interface GigabitEthernet4/0/1.2
 encapsulation dot1Q 2
 ip vrf forwarding red2
 ip address 22.2.0.1 255.255.0.0
 ip pim sparse-mode
 ip igmp version 3
!
router ospf 1
 router-id 2.2.2.2
 network 2.2.2.2 0.0.0.0 area 0
!
router bgp 100
 bgp log-neighbor-changes
 neighbor 1.1.1.1 remote-as 100
 neighbor 1.1.1.1 update-source Loopback1
 neighbor 3.3.3.3 remote-as 100
 neighbor 3.3.3.3 update-source Loopback1
 neighbor 4.4.4.4 remote-as 100
 neighbor 4.4.4.4 update-source Loopback1
 !
 address-family ipv4
  neighbor 1.1.1.1 activate
  neighbor 3.3.3.3 activate
  neighbor 4.4.4.4 activate
  no auto-summary
 exit-address-family
 !
 address-family vpnv4
  neighbor 1.1.1.1 activate
  neighbor 1.1.1.1 send-community both
  neighbor 3.3.3.3 activate
  neighbor 3.3.3.3 send-community both
 exit-address-family
 !
 address-family ipv4 mdt
  neighbor 1.1.1.1 activate
  neighbor 1.1.1.1 send-community both
  neighbor 3.3.3.3 activate
  neighbor 3.3.3.3 send-community both
 exit-address-family
 !
 address-family ipv4 vrf red2
  redistribute static
```

```
  redistribute connected
  neighbor 1.1.1.1 remote-as 100
  neighbor 1.1.1.1 activate
  neighbor 1.1.1.1 send-community both
  neighbor 3.3.3.3 remote-as 100
  neighbor 3.3.3.3 activate
  neighbor 3.3.3.3 send-community both
 exit-address-family
!
ip pim vrf red2 ssm default
!
```

Configuration on PE3 Router (Receiver PE)

```
ip vrf red2
 rd 10:2
 vpn id 10:2
 mdt default mpls MLDP 4.4.4.4
 mdt data mpls MLDP 100
 mdt data threshold 20
 route-target export 10:2
 route-target import 10:2
!
ip multicast-routing
ip multicast-routing vrf red2
!
interface Loopback1
 ip address 3.3.3.3 255.255.255.255
 ip pim sparse-mode
!
interface Loopback102
 ip vrf forwarding red2
 ip address 103.2.0.2 255.255.255.255
 ip pim sparse-mode
!
interface GigabitEthernet3/2/0.2
 encapsulation dot1Q 2
 ip vrf forwarding red2
 ip address 32.2.0.1 255.255.0.0
 ip pim sparse-mode
 ip igmp version 3
!
interface GigabitEthernet3/2/1
 ip address 30.1.1.1 255.255.255.0
 ip ospf 1 area 0
 load-interval 30
 negotiation auto
 mpls ip
 mpls label protocol ldp
!
router ospf 1
 router-id 3.3.3.3
 network 3.3.3.3 0.0.0.0 area 0
!
router bgp 100
 bgp log-neighbor-changes
 neighbor 1.1.1.1 remote-as 100
 neighbor 1.1.1.1 update-source Loopback1
 neighbor 2.2.2.2 remote-as 100
 neighbor 2.2.2.2 update-source Loopback1
 neighbor 4.4.4.4 remote-as 100
 neighbor 4.4.4.4 update-source Loopback1
 !
 address-family ipv4
```

```
 neighbor 1.1.1.1 activate
 neighbor 2.2.2.2 activate
 neighbor 4.4.4.4 activate
 no auto-summary
exit-address-family
!
address-family vpnv4
 neighbor 1.1.1.1 activate
 neighbor 1.1.1.1 send-community both
 neighbor 2.2.2.2 activate
 neighbor 2.2.2.2 send-community both
exit-address-family
!
address-family ipv4 mdt
 neighbor 1.1.1.1 activate
 neighbor 1.1.1.1 send-community both
 neighbor 2.2.2.2 activate
 neighbor 2.2.2.2 send-community both
exit-address-family
!
address-family ipv4 vrf red2
 redistribute static
 redistribute connected
 neighbor 1.1.1.1 remote-as 100
 neighbor 1.1.1.1 activate
 neighbor 1.1.1.1 send-community both
 neighbor 2.2.2.2 remote-as 100
 neighbor 2.2.2.2 activate
 neighbor 2.2.2.2 send-community both
exit-address-family
!
ip pim vrf red2 ssm default
!
```

Configuration Example Using SM Mode

These examples are based on the topology shown in Figure 18-1. Consider these scenarios while configuring MLDP MVPN using SSM mode:

- MLDP MVPN Extranet SSC
- MLDP MVPN Extranet RSC
- MLDP MVPN Intranet

MLDP MVPN Extranet SSC

Configuration on PE1 Router (Source PE)

```
ip vrf red2
 rd 10:2
 vpn id 10:2
 mdt default mpls MLDP 4.4.4.4
 mdt data mpls MLDP 100
 mdt data threshold 20
 route-target export 10:2
 route-target import 10:2
!
ip vrf red3
 rd 10:3
 vpn id 10:3
 mdt default mpls MLDP 4.4.4.4
```

```
 mdt data mpls MLDP 100
 mdt data threshold 20
 route-target export 10:3
 route-target import 10:3
!
ip multicast-routing
ip multicast-routing vrf red2
ip multicast-routing vrf red3

interface Loopback1
 ip address 1.1.1.1 255.255.255.255
 ip pim sparse-mode
!
interface Loopback102
 ip vrf forwarding red2
 ip address 101.2.0.2 255.255.255.255
 ip pim sparse-mode
!
interface Loopback103
 ip vrf forwarding red3
 ip address 101.3.0.2 255.255.255.255
 ip pim sparse-mode

interface GigabitEthernet1/22.2
 encapsulation dot1Q 2
 ip vrf forwarding red2
 ip address 12.2.0.1 255.255.0.0
 ip pim sparse-mode
!
interface TenGigabitEthernet8/1
 ip address 10.1.1.1 255.255.255.0
 ip ospf 1 area 0
 load-interval 30
 mpls ip
 mpls label protocol ldp

router ospf 1
 router-id 1.1.1.1
 network 1.1.1.1 0.0.0.0 area 0
!
router bgp 100
 bgp log-neighbor-changes
 neighbor 2.2.2.2 remote-as 100
 neighbor 2.2.2.2 update-source Loopback1
 neighbor 3.3.3.3 remote-as 100
 neighbor 3.3.3.3 update-source Loopback1
 neighbor 4.4.4.4 remote-as 100
 neighbor 4.4.4.4 update-source Loopback1
 !
 address-family ipv4
  neighbor 2.2.2.2 activate
  neighbor 3.3.3.3 activate
  neighbor 4.4.4.4 activate
  no auto-summary
 exit-address-family
 !
 address-family vpnv4
  neighbor 2.2.2.2 activate
  neighbor 2.2.2.2 send-community both
  neighbor 3.3.3.3 activate
  neighbor 3.3.3.3 send-community both
 exit-address-family
 !
 address-family ipv4 mdt
```

```
  neighbor 2.2.2.2 activate
  neighbor 2.2.2.2 send-community both
  neighbor 3.3.3.3 activate
  neighbor 3.3.3.3 send-community both
 exit-address-family
 !
 address-family ipv4 vrf red2
  redistribute static
  redistribute connected
  neighbor 2.2.2.2 remote-as 100
  neighbor 2.2.2.2 activate
  neighbor 2.2.2.2 send-community both
  neighbor 3.3.3.3 remote-as 100
  neighbor 3.3.3.3 activate
  neighbor 3.3.3.3 send-community both
 exit-address-family
 !
 address-family ipv4 vrf red3
  redistribute static
  redistribute connected
  neighbor 2.2.2.2 remote-as 100
  neighbor 2.2.2.2 activate
  neighbor 2.2.2.2 send-community both
  neighbor 3.3.3.3 remote-as 100
  neighbor 3.3.3.3 activate
  neighbor 3.3.3.3 send-community both
 exit-address-family

ip pim vrf red2 rp-address 11.11.11.11
ip pim vrf red3 rp-address 11.11.11.11
ip mroute vrf red3 12.2.0.0 255.255.0.0 fallback-lookup vrf red2
ip mroute vrf red3 11.11.11.11 255.255.0.0 fallback-lookup vrf red2
```

Configuration on P Router

```
interface Loopback1
 ip address 4.4.4.4 255.255.255.255

interface GigabitEthernet2/10
 ip address 20.1.1.2 255.255.255.0
 ip ospf 1 area 0
 load-interval 30
 mpls ip
 mpls label protocol ldp

interface GigabitEthernet2/20
 ip address 30.1.1.2 255.255.255.0
 ip ospf 1 area 0
 mpls ip
 mpls label protocol ldp

interface TenGigabitEthernet4/0/0
 ip address 10.1.1.2 255.255.255.0
 ip ospf 1 area 0
 load-interval 30
 mpls ip
 mpls label protocol ldp

router ospf 1
 router-id 4.4.4.4
 network 4.4.4.4 0.0.0.0 area 0
!
router bgp 100
```

```
 bgp log-neighbor-changes
 neighbor 1.1.1.1 remote-as 100
 neighbor 2.2.2.2 remote-as 100
 neighbor 3.3.3.3 remote-as 100
 !
 address-family ipv4
  neighbor 1.1.1.1 activate
  neighbor 2.2.2.2 activate
  neighbor 3.3.3.3 activate
  no auto-summary
 exit-address-family
```

Configuration on PE2 Router (Receiver PE)

```
ip vrf red3
 rd 10:3
 vpn id 10:3
 mdt default mpls MLDP 4.4.4.4
 mdt data mpls MLDP 100
 mdt data threshold 20
 route-target export 10:3
 route-target import 10:3
!
ip multicast-routing
ip multicast-routing vrf red3

interface Loopback1
 ip address 2.2.2.2 255.255.255.255
 ip pim sparse-mode
!
interface Loopback103
 ip vrf forwarding red3
 ip address 102.3.0.2 255.255.255.255
 ip pim sparse-mode
!
interface GigabitEthernet4/0/0
 ip address 20.1.1.1 255.255.255.0
 ip ospf 1 area 0
 load-interval 30
 negotiation auto
 mpls ip
 mpls label protocol ldp
!
interface GigabitEthernet4/0/1.3
 encapsulation dot1Q 3
 ip vrf forwarding red3
 ip address 22.2.0.1 255.255.0.0
 ip pim sparse-mode
!
router ospf 1
 router-id 2.2.2.2
 network 2.2.2.2 0.0.0.0 area 0
!
router bgp 100
 bgp log-neighbor-changes
 neighbor 1.1.1.1 remote-as 100
 neighbor 1.1.1.1 update-source Loopback1
 neighbor 3.3.3.3 remote-as 100
 neighbor 3.3.3.3 update-source Loopback1
 neighbor 4.4.4.4 remote-as 100
 neighbor 4.4.4.4 update-source Loopback1
 !
 address-family ipv4
```

```
  neighbor 1.1.1.1 activate
  neighbor 3.3.3.3 activate
  neighbor 4.4.4.4 activate
  no auto-summary
 exit-address-family
 !
 address-family vpnv4
  neighbor 1.1.1.1 activate
  neighbor 1.1.1.1 send-community both
  neighbor 3.3.3.3 activate
  neighbor 3.3.3.3 send-community both
 exit-address-family
 !
 address-family ipv4 mdt
  neighbor 1.1.1.1 activate
  neighbor 1.1.1.1 send-community both
  neighbor 3.3.3.3 activate
  neighbor 3.3.3.3 send-community both
 exit-address-family
 !
 address-family ipv4 vrf red3
  redistribute static
  redistribute connected
  neighbor 1.1.1.1 remote-as 100
  neighbor 1.1.1.1 activate
  neighbor 1.1.1.1 send-community both
  neighbor 3.3.3.3 remote-as 100
  neighbor 3.3.3.3 activate
  neighbor 3.3.3.3 send-community both
 exit-address-family
!
ip pim vrf red3 rp-address 11.11.11.11
ip mroute vrf red3 12.2.0.0 255.255.0.0 101.3.0.2
```

Configuraton on PE3 Router (Receiver PE)

```
ip vrf red3
 rd 10:3
 vpn id 10:3
 mdt default mpls MLDP 4.4.4.4
 mdt data mpls MLDP 100
 mdt data threshold 20
 route-target export 10:3
 route-target import 10:3
!
ip multicast-routing
ip multicast-routing vrf red3
!
interface Loopback1
 ip address 3.3.3.3 255.255.255.255
 ip pim sparse-mode
!
interface Loopback103
 ip vrf forwarding red3
 ip address 103.3.0.2 255.255.255.255
 ip pim sparse-mode
!
interface GigabitEthernet3/2/0.3
 encapsulation dot1Q 3
 ip vrf forwarding red3
 ip address 32.2.0.1 255.255.0.0
 ip pim sparse-mode
 ip igmp version 3
!
```

```
interface GigabitEthernet3/2/1
 ip address 30.1.1.1 255.255.255.0
 ip ospf 1 area 0
 load-interval 30
 negotiation auto
 mpls ip
 mpls label protocol ldp
!
router ospf 1
 router-id 3.3.3.3
 network 3.3.3.3 0.0.0.0 area 0
!
router bgp 100
 bgp log-neighbor-changes
 neighbor 1.1.1.1 remote-as 100
 neighbor 1.1.1.1 update-source Loopback1
 neighbor 2.2.2.2 remote-as 100
 neighbor 2.2.2.2 update-source Loopback1
 neighbor 4.4.4.4 remote-as 100
 neighbor 4.4.4.4 update-source Loopback1
 !
 address-family ipv4
  neighbor 1.1.1.1 activate
  neighbor 2.2.2.2 activate
  neighbor 4.4.4.4 activate
  no auto-summary
 exit-address-family
 !
 address-family vpnv4
  neighbor 1.1.1.1 activate
  neighbor 1.1.1.1 send-community both
  neighbor 2.2.2.2 activate
  neighbor 2.2.2.2 send-community both
 exit-address-family
 !
 address-family ipv4 mdt
  neighbor 1.1.1.1 activate
  neighbor 1.1.1.1 send-community both
  neighbor 2.2.2.2 activate
  neighbor 2.2.2.2 send-community both
 exit-address-family
 !
 address-family ipv4 vrf red3
  redistribute static
  redistribute connected
  neighbor 1.1.1.1 remote-as 100
  neighbor 1.1.1.1 activate
  neighbor 1.1.1.1 send-community both
  neighbor 2.2.2.2 remote-as 100
  neighbor 2.2.2.2 activate
  neighbor 2.2.2.2 send-community both
 exit-address-family
!
ip pim vrf red3 rp-address 11.11.11.11
ip mroute vrf red3 12.2.0.0 255.255.0.0 101.3.0.2
```

MLDP MVPN Extranet RSC

Configuration on PE1 Router (Source PE)

```
ip vrf red2
 rd 10:2
 vpn id 10:2
```

```
 mdt default mpls MLDP 4.4.4.4
 mdt data mpls MLDP 100
 mdt data threshold 20
 route-target export 10:2
 route-target import 10:2
!
ip multicast-routing
ip multicast-routing vrf red2
!
interface Loopback1
 ip address 1.1.1.1 255.255.255.255
 ip pim sparse-mode
!
interface Loopback102
 ip vrf forwarding red2
 ip address 101.2.0.2 255.255.255.255
 ip pim sparse-mode
!
interface GigabitEthernet1/22.2
 encapsulation dot1Q 2
 ip vrf forwarding red2
 ip address 12.2.0.1 255.255.0.0
 ip pim sparse-mode
!
interface TenGigabitEthernet8/1
 ip address 10.1.1.1 255.255.255.0
 ip ospf 1 area 0
 load-interval 30
 mpls ip
 mpls label protocol ldp
!
router ospf 1
 router-id 1.1.1.1
 network 1.1.1.1 0.0.0.0 area 0
!
router bgp 100
 bgp log-neighbor-changes
 neighbor 2.2.2.2 remote-as 100
 neighbor 2.2.2.2 update-source Loopback1
 neighbor 3.3.3.3 remote-as 100
 neighbor 3.3.3.3 update-source Loopback1
 neighbor 4.4.4.4 remote-as 100
 neighbor 4.4.4.4 update-source Loopback1
 !
 address-family ipv4
  neighbor 2.2.2.2 activate
  neighbor 3.3.3.3 activate
  neighbor 4.4.4.4 activate
  no auto-summary
 exit-address-family
 !
 address-family vpnv4
  neighbor 2.2.2.2 activate
  neighbor 2.2.2.2 send-community both
  neighbor 3.3.3.3 activate
  neighbor 3.3.3.3 send-community both
 exit-address-family
 !
 address-family ipv4 mdt
  neighbor 2.2.2.2 activate
  neighbor 2.2.2.2 send-community both
  neighbor 3.3.3.3 activate
  neighbor 3.3.3.3 send-community both
 exit-address-family
```

```
 !
 address-family ipv4 vrf red2
  redistribute static
  redistribute connected
  neighbor 2.2.2.2 remote-as 100
  neighbor 2.2.2.2 activate
  neighbor 2.2.2.2 send-community both
  neighbor 3.3.3.3 remote-as 100
  neighbor 3.3.3.3 activate
  neighbor 3.3.3.3 send-community both
 exit-address-family
!
ip pim vrf red2 rp-address 11.11.11.11
```

Configuration on P Router (Core Router)

```
interface Loopback1
 ip address 4.4.4.4 255.255.255.255
!
interface GigabitEthernet2/10
 ip address 20.1.1.2 255.255.255.0
 ip ospf 1 area 0
 load-interval 30
 mpls ip
 mpls label protocol ldp
!
interface GigabitEthernet2/20
 ip address 30.1.1.2 255.255.255.0
 ip ospf 1 area 0
 mpls ip
 mpls label protocol ldp
!
interface TenGigabitEthernet4/0/0
 ip address 10.1.1.2 255.255.255.0
 ip ospf 1 area 0
 load-interval 30
 mpls ip
 mpls label protocol ldp
 mls qos trust dscp
!
router ospf 1
 router-id 4.4.4.4
 network 4.4.4.4 0.0.0.0 area 0
!
router bgp 100
 bgp log-neighbor-changes
 neighbor 1.1.1.1 remote-as 100
 neighbor 2.2.2.2 remote-as 100
 neighbor 3.3.3.3 remote-as 100
 !
 address-family ipv4
  neighbor 1.1.1.1 activate
  neighbor 2.2.2.2 activate
  neighbor 3.3.3.3 activate
  no auto-summary
 exit-address-family
!
```

Configuration ond PE2 Router (Receiver PE)

```
ip vrf red2
```

```
 rd 10:2
 vpn id 10:2
 mdt default mpls MLDP 4.4.4.4
 mdt data mpls MLDP 100
 mdt data threshold 20
 route-target export 10:2
 route-target import 10:2
!
ip vrf red3
 rd 10:3
 vpn id 10:3
 mdt default mpls MLDP 4.4.4.4
 mdt data mpls MLDP 100
 mdt data threshold 20
 route-target export 10:3
 route-target import 10:3
!
ip multicast-routing
ip multicast-routing vrf red3
ip multicast-routing vrf red2
!
interface Loopback1
 ip address 2.2.2.2 255.255.255.255
 ip pim sparse-mode
!
interface Loopback102
 ip vrf forwarding red2
 ip address 102.2.0.2 255.255.255.255
 ip pim sparse-mode
!
interface Loopback103
 ip vrf forwarding red3
 ip address 102.3.0.2 255.255.255.255
 ip pim sparse-mode
!
interface GigabitEthernet4/0/0
 ip address 20.1.1.1 255.255.255.0
 ip ospf 1 area 0
 load-interval 30
 negotiation auto
 mpls ip
 mpls label protocol ldp
!
interface GigabitEthernet4/0/1.3
 encapsulation dot1Q 3
 ip vrf forwarding red3
 ip address 22.2.0.1 255.255.0.0
 ip pim sparse-mode
!
router ospf 1
 router-id 2.2.2.2
 network 2.2.2.2 0.0.0.0 area 0
!
router bgp 100
 bgp log-neighbor-changes
 neighbor 1.1.1.1 remote-as 100
 neighbor 1.1.1.1 update-source Loopback1
 neighbor 3.3.3.3 remote-as 100
 neighbor 3.3.3.3 update-source Loopback1
 neighbor 4.4.4.4 remote-as 100
 neighbor 4.4.4.4 update-source Loopback1
 !
 address-family ipv4
  neighbor 1.1.1.1 activate
```

```
 neighbor 3.3.3.3 activate
 neighbor 4.4.4.4 activate
 no auto-summary
exit-address-family
!
address-family vpnv4
 neighbor 1.1.1.1 activate
 neighbor 1.1.1.1 send-community both
 neighbor 3.3.3.3 activate
 neighbor 3.3.3.3 send-community both
exit-address-family
!
address-family ipv4 mdt
 neighbor 1.1.1.1 activate
 neighbor 1.1.1.1 send-community both
 neighbor 3.3.3.3 activate
 neighbor 3.3.3.3 send-community both
exit-address-family
!
address-family ipv4 vrf red2
 redistribute static
 redistribute connected
 neighbor 1.1.1.1 remote-as 100
 neighbor 1.1.1.1 activate
 neighbor 1.1.1.1 send-community both
 neighbor 3.3.3.3 remote-as 100
 neighbor 3.3.3.3 activate
 neighbor 3.3.3.3 send-community both
exit-address-family
!
address-family ipv4 vrf red3
 redistribute static
 redistribute connected
 neighbor 1.1.1.1 remote-as 100
 neighbor 1.1.1.1 activate
 neighbor 1.1.1.1 send-community both
 neighbor 3.3.3.3 remote-as 100
 neighbor 3.3.3.3 activate
 neighbor 3.3.3.3 send-community both
exit-address-family
!
ip pim vrf red2 rp-address 11.11.11.11
ip pim vrf red3 rp-address 11.11.11.11
ip mroute vrf red3 12.2.0.0 255.255.0.0 fallback-lookup vrf red2
ip mroute vrf red3 11.11.11.11 255.255.255.255 fallback-lookup vrf red2
```

Configuration on PE3 Router (Receiver PE)

```
ip vrf red2
 rd 10:2
 vpn id 10:2
 mdt default mpls MLDP 4.4.4.4
 mdt data mpls MLDP 100
 mdt data threshold 20
 route-target export 10:2
 route-target import 10:2
!
ip vrf red3
 rd 10:3
 vpn id 10:3
 mdt default mpls MLDP 4.4.4.4
 mdt data mpls MLDP 100
```

```
 mdt data threshold 20
 route-target export 10:3
 route-target import 10:3
!
ip multicast-routing
ip multicast-routing vrf red3
ip multicast-routing vrf red2
!
interface Loopback1
 ip address 3.3.3.3 255.255.255.255
 ip pim sparse-mode
!
interface Loopback102
 ip vrf forwarding red2
 ip address 103.2.0.2 255.255.255.255
 ip pim sparse-mode
!
interface Loopback103
 ip vrf forwarding red3
 ip address 103.3.0.2 255.255.255.255
 ip pim sparse-mode
!
interface GigabitEthernet3/2/0.3
 encapsulation dot1Q 3
 ip vrf forwarding red3
 ip address 32.2.0.1 255.255.0.0
 ip pim sparse-mode
 ip igmp version 3
!
interface GigabitEthernet3/2/1
 ip address 30.1.1.1 255.255.255.0
 ip ospf 1 area 0
 load-interval 30
 negotiation auto
 mpls ip
 mpls label protocol ldp
!
router ospf 1
 router-id 3.3.3.3
 network 3.3.3.3 0.0.0.0 area 0
!
router bgp 100
 bgp log-neighbor-changes
 neighbor 1.1.1.1 remote-as 100
 neighbor 1.1.1.1 update-source Loopback1
 neighbor 2.2.2.2 remote-as 100
 neighbor 2.2.2.2 update-source Loopback1
 neighbor 4.4.4.4 remote-as 100
 neighbor 4.4.4.4 update-source Loopback1
 !
 address-family ipv4
  neighbor 1.1.1.1 activate
  neighbor 2.2.2.2 activate
  neighbor 4.4.4.4 activate
  no auto-summary
 exit-address-family
 !
 address-family vpnv4
  neighbor 1.1.1.1 activate
  neighbor 1.1.1.1 send-community both
  neighbor 2.2.2.2 activate
  neighbor 2.2.2.2 send-community both
 exit-address-family
 !
```

```
 address-family ipv4 mdt
  neighbor 1.1.1.1 activate
  neighbor 1.1.1.1 send-community both
  neighbor 2.2.2.2 activate
  neighbor 2.2.2.2 send-community both
 exit-address-family
 !
 address-family ipv4 vrf red2
  redistribute static
  redistribute connected
  neighbor 1.1.1.1 remote-as 100
  neighbor 1.1.1.1 activate
  neighbor 1.1.1.1 send-community both
  neighbor 2.2.2.2 remote-as 100
  neighbor 2.2.2.2 activate
  neighbor 2.2.2.2 send-community both
 exit-address-family
 !
 address-family ipv4 vrf red3
  redistribute static
  redistribute connected
  neighbor 1.1.1.1 remote-as 100
  neighbor 1.1.1.1 activate
  neighbor 1.1.1.1 send-community both
  neighbor 2.2.2.2 remote-as 100
  neighbor 2.2.2.2 activate
  neighbor 2.2.2.2 send-community both
 exit-address-family
!
ip pim vrf red2 rp-address 11.11.11.11
ip pim vrf red3 rp-address 11.11.11.11
ip mroute vrf red3 12.2.0.0 255.255.0.0 fallback-lookup vrf red2
ip mroute vrf red3 11.11.11.11 255.255.255.255 fallback-lookup vrf red2
```

MLDP MVPN Intranet

Configuration ond PE1 Router (Source PE)

```
ip vrf red2
 rd 10:2
 vpn id 10:2
 mdt default mpls MLDP 4.4.4.4
 mdt data mpls MLDP 100
 mdt data threshold 20
 route-target export 10:2
 route-target import 10:2
!
ip multicast-routing
ip multicast-routing vrf red2
!
interface Loopback1
 ip address 1.1.1.1 255.255.255.255
 ip pim sparse-mode
!
interface Loopback102
 ip vrf forwarding red2
 ip address 101.2.0.2 255.255.255.255
 ip pim sparse-mode
!
interface GigabitEthernet1/22.2
 encapsulation dot1Q 2
 ip vrf forwarding red2
 ip address 12.2.0.1 255.255.0.0
```

```
 ip pim sparse-mode
!
interface TenGigabitEthernet8/1
 ip address 10.1.1.1 255.255.255.0
 ip ospf 1 area 0
 load-interval 30
 mpls ip
 mpls label protocol ldp
!
router ospf 1
 router-id 1.1.1.1
 network 1.1.1.1 0.0.0.0 area 0
!
router bgp 100
 bgp log-neighbor-changes
 neighbor 2.2.2.2 remote-as 100
 neighbor 2.2.2.2 update-source Loopback1
 neighbor 3.3.3.3 remote-as 100
 neighbor 3.3.3.3 update-source Loopback1
 neighbor 4.4.4.4 remote-as 100
 neighbor 4.4.4.4 update-source Loopback1
 !
 address-family ipv4
  neighbor 2.2.2.2 activate
  neighbor 3.3.3.3 activate
  neighbor 4.4.4.4 activate
  no auto-summary
 exit-address-family
 !
 address-family vpnv4
  neighbor 2.2.2.2 activate
  neighbor 2.2.2.2 send-community both
  neighbor 3.3.3.3 activate
  neighbor 3.3.3.3 send-community both
 exit-address-family
 !
 address-family ipv4 mdt
  neighbor 2.2.2.2 activate
  neighbor 2.2.2.2 send-community both
  neighbor 3.3.3.3 activate
  neighbor 3.3.3.3 send-community both
 exit-address-family
 !
 address-family ipv4 vrf red2
  redistribute static
  redistribute connected
  neighbor 2.2.2.2 remote-as 100
  neighbor 2.2.2.2 activate
  neighbor 2.2.2.2 send-community both
  neighbor 3.3.3.3 remote-as 100
  neighbor 3.3.3.3 activate
  neighbor 3.3.3.3 send-community both
 exit-address-family
!
ip pim vrf red2 rp-address 11.11.11.11
```

Configuration ond P Router (Core Router)

```
interface Loopback1
 ip address 4.4.4.4 255.255.255.255
!
interface GigabitEthernet2/10
 ip address 20.1.1.2 255.255.255.0
```

```
 ip ospf 1 area 0
 load-interval 30
 mpls ip
 mpls label protocol ldp
!
interface GigabitEthernet2/20
 ip address 30.1.1.2 255.255.255.0
 ip ospf 1 area 0
 mpls ip
 mpls label protocol ldp
!
interface TenGigabitEthernet4/0/0
 ip address 10.1.1.2 255.255.255.0
 ip ospf 1 area 0
 load-interval 30
 mpls ip
 mpls label protocol ldp
 mls qos trust dscp
!
router ospf 1
 router-id 4.4.4.4
 network 4.4.4.4 0.0.0.0 area 0
!
router bgp 100
 bgp log-neighbor-changes
 neighbor 1.1.1.1 remote-as 100
 neighbor 2.2.2.2 remote-as 100
 neighbor 3.3.3.3 remote-as 100
 !
 address-family ipv4
  neighbor 1.1.1.1 activate
  neighbor 2.2.2.2 activate
  neighbor 3.3.3.3 activate
  no auto-summary
 exit-address-family
!
```

Configuration on PE2 Router (Receiver PE)

```
ip vrf red2
 rd 10:2
 vpn id 10:2
 mdt default mpls MLDP 4.4.4.4
 mdt data mpls MLDP 100
 mdt data threshold 20
 route-target export 10:2
 route-target import 10:2
!
ip multicast-routing
ip multicast-routing vrf red2
!
interface Loopback1
 ip address 2.2.2.2 255.255.255.255
 ip pim sparse-mode
!
interface Loopback102
 ip vrf forwarding red2
 ip address 102.2.0.2 255.255.255.255
 ip pim sparse-mode
!
interface GigabitEthernet4/0/0
 ip address 20.1.1.1 255.255.255.0
 ip ospf 1 area 0
```

```
 load-interval 30
 negotiation auto
 mpls ip
 mpls label protocol ldp
!
interface GigabitEthernet4/0/1.2
 encapsulation dot1Q 2
 ip vrf forwarding red2
 ip address 22.2.0.1 255.255.0.0
 ip pim sparse-mode
 ip igmp version 3
!
router ospf 1
 router-id 2.2.2.2
 network 2.2.2.2 0.0.0.0 area 0
!
router bgp 100
 bgp log-neighbor-changes
 neighbor 1.1.1.1 remote-as 100
 neighbor 1.1.1.1 update-source Loopback1
 neighbor 3.3.3.3 remote-as 100
 neighbor 3.3.3.3 update-source Loopback1
 neighbor 4.4.4.4 remote-as 100
 neighbor 4.4.4.4 update-source Loopback1
 !
 address-family ipv4
  neighbor 1.1.1.1 activate
  neighbor 3.3.3.3 activate
  neighbor 4.4.4.4 activate
  no auto-summary
 exit-address-family
 !
 address-family vpnv4
  neighbor 1.1.1.1 activate
  neighbor 1.1.1.1 send-community both
  neighbor 3.3.3.3 activate
  neighbor 3.3.3.3 send-community both
 exit-address-family
 !
 address-family ipv4 mdt
  neighbor 1.1.1.1 activate
  neighbor 1.1.1.1 send-community both
  neighbor 3.3.3.3 activate
  neighbor 3.3.3.3 send-community both
 exit-address-family
 !
 address-family ipv4 vrf red2
  redistribute static
  redistribute connected
  neighbor 1.1.1.1 remote-as 100
  neighbor 1.1.1.1 activate
  neighbor 1.1.1.1 send-community both
  neighbor 3.3.3.3 remote-as 100
  neighbor 3.3.3.3 activate
  neighbor 3.3.3.3 send-community both
 exit-address-family
!
ip pim vrf red2 rp-address 11.11.11.11
!
```

Configuration on PE3 Router (Receiver PE)

```
ip vrf red2
```

```
 rd 10:2
 vpn id 10:2
 mdt default mpls MLDP 4.4.4.4
 mdt data mpls MLDP 100
 mdt data threshold 20
 route-target export 10:2
 route-target import 10:2
!
ip multicast-routing
ip multicast-routing vrf red2
!
interface Loopback1
 ip address 3.3.3.3 255.255.255.255
 ip pim sparse-mode
!
interface Loopback102
 ip vrf forwarding red2
 ip address 103.2.0.2 255.255.255.255
 ip pim sparse-mode
!
interface GigabitEthernet3/2/0.2
 encapsulation dot1Q 2
 ip vrf forwarding red2
 ip address 32.2.0.1 255.255.0.0
 ip pim sparse-mode
 ip igmp version 3
!
interface GigabitEthernet3/2/1
 ip address 30.1.1.1 255.255.255.0
 ip ospf 1 area 0
 load-interval 30
 negotiation auto
 mpls ip
 mpls label protocol ldp
!
router ospf 1
 router-id 3.3.3.3
 network 3.3.3.3 0.0.0.0 area 0
!
router bgp 100
 bgp log-neighbor-changes
 neighbor 1.1.1.1 remote-as 100
 neighbor 1.1.1.1 update-source Loopback1
 neighbor 2.2.2.2 remote-as 100
 neighbor 2.2.2.2 update-source Loopback1
 neighbor 4.4.4.4 remote-as 100
 neighbor 4.4.4.4 update-source Loopback1
 !
 address-family ipv4
  neighbor 1.1.1.1 activate
  neighbor 2.2.2.2 activate
  neighbor 4.4.4.4 activate
  no auto-summary
 exit-address-family
 !
 address-family vpnv4
  neighbor 1.1.1.1 activate
  neighbor 1.1.1.1 send-community both
  neighbor 2.2.2.2 activate
  neighbor 2.2.2.2 send-community both
 exit-address-family
 !
 address-family ipv4 mdt
  neighbor 1.1.1.1 activate
```

```
  neighbor 1.1.1.1 send-community both
  neighbor 2.2.2.2 activate
  neighbor 2.2.2.2 send-community both
 exit-address-family
 !
 address-family ipv4 vrf red2
  redistribute static
  redistribute connected
  neighbor 1.1.1.1 remote-as 100
  neighbor 1.1.1.1 activate
  neighbor 1.1.1.1 send-community both
  neighbor 2.2.2.2 remote-as 100
  neighbor 2.2.2.2 activate
  neighbor 2.2.2.2 send-community both
 exit-address-family
!
ip pim vrf red2 rp-address 11.11.11.11
!
```

Troubleshooting LSM MLDP based MVPN Support

Use these debug commands to troubleshoot the LSM MLDP based MVPN support on the Cisco ASR 1000 Series Aggregation Services Routers.

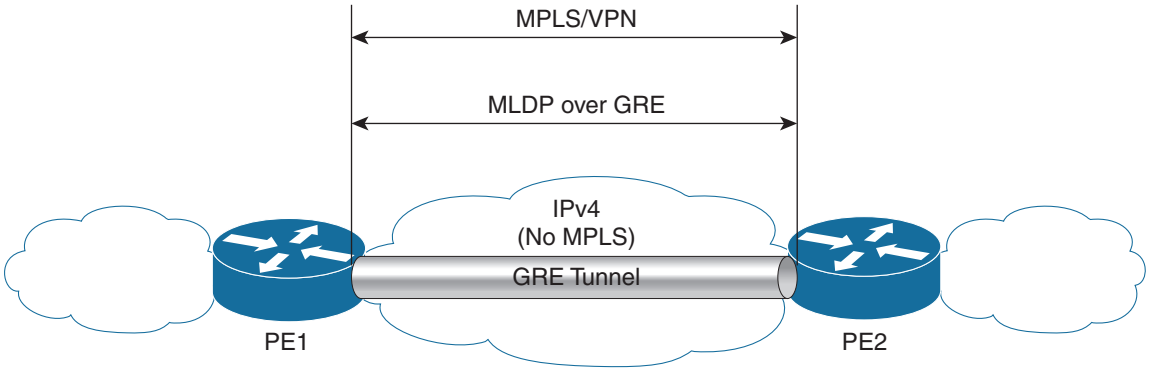
Command	Purpose
debug mpls MLDP packet **debug mpls MLDP neighbor** **debug mpls MLDP all**	Used for MLDP debugging [RP].
debug ip igmp vrf blue	Used for IGMP debugs.
debug ip pim vrf blue hello **debug ip pim vrf blue timer** **debug ip pim vrf blue bsr** **debug ip pim vrf blue auto-rp**	Used for PIM debugs [RP].
debug mpls infra lfd mfi	Used for IOS layer debugs.
deb pl so mpls	Used for IOSD shim layer debugs.
configure terminal **platform trace [run|boot] slot [f0|f1|r0|r1] bay 0 process for mod cef level [debug|verbose]** **end**	Used for FMAN-RP/FMAN-FP.
debug platform hardware qfp active feature cef-mpls client mpls all	Used for QFP client.
debug platform hardware qfp active feature cef-mpls datapath mpls all	Used for QFP server.

MVPN MLDP over GRE

The Multicast Label Distribution Protocol- based Multicast VPN (MVPN) feature supports IPv4 and IPv6 multicast traffic over a Multi-Protocol Label Switching (MPLS) network. But a large part of the network infrastructure is still IP network, and the legacy IP network does not support MPLS. The existing MPLS over Generic Routing Encapsulation (GRE) feature provides a mechanism for tunneling MPLS packets over a non-MPLS network by creating a GRE tunnel across the IP network and bridging the separated MPLS networks. However, the existing MPLS over GRE feature does not support MPLS multicast traffic. The MVPN MLDP over GRE feature provides a solution by supporting encapsulating MPLS multicast traffic in the GRE tunnel.

Figure 18-2 shows a sample configuration for MVPN Multicast Label Distribution Protocol over GRE using the PE-PE network topology.

Figure 18-2 MVPN MLDP over GRE with PE-PE Network Topology



Prerequisites for MVPN MLDP over GRE

- Ensure that MPLS Virtual Private Network (MVPN) is configured and working properly. For information about setting up MPLS VPNs, see:

  http://www.cisco.com/en/US/docs/ios-xml/ios/mp_l3_vpns/configuration/xe-3s/asr1000/mp-cfg-layer3-vpn.html

- Ensure that Multiprotocol Border Gateway Protocol (MP-BGP) is configured and working properly. For more information about configuring (MP-BGP), see:

  http://www.cisco.com/en/US/docs/ios-xml/ios/mp_l3_vpns/configuration/xe-3s/asr1000/mp-bgp-mpls-vpn.html

Restrictions for MVPN MLDP over GRE

The following are the restrictions that you will encounter while configuring the MVPN MLDP over GRE feature:

- MVPN MLDP over GRE supports only IPv4 GRE.

- MVPN MLDP over GRE supports IPv4 and IPv6 multicast traffic.

Configuring MVPN MLDP over GRE

Complete these steps to configure MVPN MLDP over GRE with PE-to-PE topology. You should perform these steps on both the PE routers.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **mpls MLDP**
4. **vrf definition** *vrf-name*
5. **rd** *route-distinguisher*
6. **vpn id** *vpn_id*
7. **address-family ipv4**
8. **mdt default mpls MLDP** *root-node*
9. **mdt data mpls MLDP** *number-of-data-MDTs*
10. **mdt data threshold** *bandwidth*
11. **route-target** [**import** | **export** | **both**] *route-target-ext-community*
12. **route-target [import** | **export** | **both**] *route-target-ext-community*
13. **exit**
14. **address-family ipv6**
15. **mdt default mpls MLDP** *root-node*
16. **mdt default mpls MLDP** *root-node*
17. **mdt data mpls MLDP** *number-of-data-MDTs*
18. **mdt data threshold** *bandwidth*
19. **route-target** [**import** | **export** | **both**] *route-target-ext-community*
20. **route-target [import** | **export** | **both**] *route-target-ext-community*
21. **exit**
22. **exit**
23. **interface** *name*
24. **vrf forwarding** *vrf-name*
25. **ip address** *ip-address subnet-mask*
26. **ip pim sparse-mode**
27. **ipv6 address** *ipv6-address*
28. **ospfv3100 ipv6 area** *0*
29. **ip multicast-routing vrf** *vrf-name* **distributed**
30. **ip multicast-routing vrf** *vrf-name*
31. **exit**

32. **end**

DETAILED STEPS

	Command	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables the privileged EXEC mode. Enter your password when prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters the global configuration mode.
Step 3	**mpls MLDP** **Example:** `Router(config)# mpls MLDP`	Enables MPLS MLDP support. **Note** The **mpls MLDP** command is configured by default. To disable MPLS MLDP, use the **no mpls MLDP** command.
Step 4	**vrf definition** *vrf-name* **Example:** `Router(config)# vrf definition blue`	Defines the VPN routing instance by assigning a VRF name, and enters the VRF configuration mode. *vrf-name*—Name assigned to a VRF.
Step 5	**rd** *route-distinguisher* **Example:** `Router(config-vrf)# rd 200:2`	Creates routing and forwarding tables. *route-distinguisher*— Specifies the 8-byte value to create a VPN prefix. You can enter a *route-distinguisher* value in either of these formats: • *16-bit autonomous system number*: Your 16-bit number Example, 200:2. • *32-bit IP address*: Your 32-bit number Example, 192.168.122.15:1.
Step 6	**vpn id** *vpn-id* **Example:** `Router(config-vrf)# vpn id 200:2`	Sets or updates a VPN identifier on a VRF.
Step 7	**address-family ipv4** **Example:** `Router(config-vrf)# address-family ipv4`	Enters the address family configuration mode using standard IP Version 4 (IPv4) address prefixes.

	Command	Purpose
Step 8	**mdt default mpls MLDP** *root-node* **Example:** `Router(config-vrf-af)# mdt default mpls MLDP 1.1.1.1`	Configures MLDP MDT for a VRF. *root-node*—The root node can be IP address of a loopback or physical interface on any router (source PE, receiver PE, or core router) in the provider network. The root node address should be accessible to all the routers in the network. The router from where signalling occurs functions as the root node. The default MDT must be configured on each PE router to enable the PE routers to receive multicast traffic for this particular MVRF. **Note** Creates the LSPVIF tunnel with the **mdt default mpls MLDP** *root-node* command.
Step 9	**mdt default mpls MLDP** *root-node* **Example:** `Router(config-vrf-af)# mdt default mpls MLDP 1.1.1.2`	Configures Root Node Redundancy. *root-node*—The root node can be IP address of a loopback or physical interface on any router (source PE, receiver PE, or core router) in the provider network. The root node address should be accessible to all the routers in the network. The router from where signaling occurs functions as the root node. The default MDT must be configured on each PE router to enable the PE routers to receive multicast traffic for this particular MVRF.
Step 10	**mdt data mpls MLDP** *number_of_data_MDTs* **Example:** `Router(config-vrf-af)# mdt data mpls MLDP 20`	Configures the MLDP data MDP.

	Command	Purpose
Step 11	**mdt data threshold** *bandwidth* **Example:** `Router(config-vrf-af)# mdt data threshold 1`	Configures the threshold value for data MDT. **Note** Bandwidth is traffic rate, in Kbps.
Step 12	**route-target export** *route-target-ext-community* **Example:** `Router(config-vrf-af)# route-target export 200:2`	Creates a route target extended community for a VRF. • **export**—Exports the routing information from the target VPN extended community. • *route-target-ext-community*—Adds the route target extended community attributes to the VRF list of import, export, or both (import and export) route target extended communities.
Step 13	**route-target import** *route-target-ext-community* **Example:** `Router(config-vrf-af)# route-target import 200:2`	Creates a route-target extended community for a VRF. • **import**—Imports the routing information from the target VPN extended community. • *route-target-ext-community*—Adds the route target extended community attributes to the VRF list of import, export, or both (import and export) route-target extended communities.
Step 14	**exit** **Example:** `Router(config-vrf-af)# exit`	Exits the address family configuration mode.
Step 15	**address-family ipv6** **Example:** `Router(config-vrf)# address-family ipv6`	Enters the address family configuration mode using standard IP Version 6 (IPv6) address prefixes.

	Command	Purpose
Step 16	**mdt default mpls MLDP** *root-node* **Example:** `Router(config-vrf-af)# mdt default mpls MLDP 1.1.1.1`	Configures MLDP MDT for a VRF. *root-node*—The root node can be IP address of a loopback or physical interface on any router (source PE, receiver PE, or core router) in the provider network. The root node address should be accessible to all the routers in the network. The router from where signalling occurs functions as the root node. The default MDT must be configured on each PE router to enable the PE routers to receive multicast traffic for this particular MVRF. **Note** Creates the LSPVIF tunnel with the **mdt default mpls MLDP** *root-node* command.
Step 17	**mdt default mpls MLDP** *root-node* **Example:** `Router(config-vrf-af)# mdt default mpls MLDP 1.1.1.2`	Configures Root Node Redundancy. *root-node*—The root node can be IP address of a loopback or physical interface on any router (source PE, receiver PE, or core router) in the provider network. The root node address should be accessible to all the routers in the network. The router from where signalling occurs functions as the root node. The default MDT must be configured on each PE router to enable the PE routers to receive multicast traffic for this particular MVRF.
Step 18	**mdt data mpls MLDP** *number_of_data_MDTs* **Example:** `Router(config-vrf-af)# mdt data mpls MLDP 20`	Configures the MLDP data MDP.

	Command	Purpose
Step 19	**mdt data threshold** *bandwidth* **Example:** `Router(config-vrf-af)# mdt data threshold 1`	Configures the threshold value for data MDT. **Note** Bandwidth is traffic rate, in Kbps.
Step 20	**route-target export** *route-target-ext-community* **Example:** `Router(config-vrf-af)# route-target export 200:2`	Creates a route target extended community for a VRF. • **export**—Exports the routing information from the target VPN extended community. • *route-target-ext-community*—Adds the route target extended community attributes to the VRF list of import, export, or both (import and export) route target extended communities.
Step 21	**route-target import** *route-target-ext-community* **Example:** `Router(config-vrf-af)# route-target import 200:2`	Creates a route-target extended community for a VRF. • **import**—Imports the routing information from the target VPN extended community. • *route-target-ext-community*—Adds the route target extended community attributes to the VRF list of import, export, or both (import and export) route-target extended communities.
Step 22	**exit** **Example:** `Router(config-vrf-af)# exit`	Exits the address family configuration mode.
Step 23	**exit** **Example:** `Router(config-if)# exit`	Exits the interface configuration mode.
Step 24	**interface** name **Example:** `Router(config)# interface gi0/0/0`	Specifies the interface name and enters the interface configuration mode.
Step 25	**vrf forwarding** *vrf-name* **Example:** `Router(config-if)# vrf forwarding blue`	Associates a VRF instance with an interface or subinterface. • *vrf-name*—Name assigned to a VRF.

	Command	Purpose
Step 26	**ip address** ip-address subnet-mask **Example:** Router(config-if)# ip address 30.2.0.1 255.255.255.0	Specifies the interface IPv4 address and subnet-mask.
Step 27	**ip pim sparse-mode** **Example:** Router(config-if)# ip pim sparse-mode	Enables sparse mode.
Step 28	**ipv6 address** ipv6-address **Example:** Router(config-if)# ipv6 address 32002:30:2::1/64	Specifies the interface IPv6 address.
Step 29	**ospfv3100 ipv6 area** 0	Enables OSPFv3 router configuration mode for the IPv6 address family.
Step 30	**end** **Example:** Router(config)# end	Ends the configuration session.
Step 31	**ip multicast-routing vrf** *vrf-name* **distributed** **Example:** Router(config)# ip multicast-routing vrf blue distributed	Enables multicast routing for the specified VRF.
Step 32	**ipv6 multicast-routing vrf** *vrf-name* **Example:** Router(config)# ipv6 multicast-routing vrf blue	Enables IPv6 multicast routing for the specified VRF.
Step 33	**exit** **Example:** Router(config-if)# exit	Exits the interface configuration mode.
Step 34	**end** **Example:** Router(config)# end	Ends the configuration session.

Example: Configuring MVPN MLDP over GRE

The following example shows how to configure MVPN MLDP over GRE:

```
Router> enable
Router# configure terminal
Router(config)# mpls MLDP
Router(config)# vrf definition blue
Router(config-vrf)# rd 200:2
Router(config-vrf)# vpn id 200:2
Router(config-vrf)# address-family ipv4
Router(config-vrf-af)# mdt default mpls MLDP 1.1.1.1
Router(config-vrf-af)# mdt default mpls MLDP 1.1.1.2
Router(config-vrf-af)# mdt data mpls MLDP 20
Router(config-vrf-af)# mdt data threshold 1
Router(config-vrf-af)# route-target export 200:2
Router(config-vrf-af)# route-target import 200:2
Router(config-vrf-af)# exit
Router(config-vrf)# address-family ipv6
Router(config-vrf-af)# mdt default mpls MLDP 1.1.1.1
Router(config-vrf-af)# mdt default mpls MLDP 1.1.1.2
Router(config-vrf-af)# mdt data mpls MLDP 20
Router(config-vrf-af)# mdt data threshold 1
Router(config-vrf-af)# route-target export 200:2
Router(config-vrf-af)# route-target import 200:2
Router(config-vrf-af)# exit
Router(config-if)# exit
Router(config)# interface gi0/0/0
Router(config-if)# vrf forwarding blue
Router(config-if)# ip address 30.2.0.1 255.255.255.0
Router(config-if)# ip pim sparse-mode
Router(config-if)# ipv6 address 32002:30:2::1/64
Router(config-if)# ospfv3100 ipv6 area 0
Router(config)# end
Router(config)# ip multicast-routing vrf blue distributed
Router(config)# ipv6 multicast-routing vrf blue
Router(config-if)# exit
Router(config)# end
```

The following example shows how to configure MVPNv4 MLDP over GRE on router PE1:

```
Router# enable
Router# configure terminal
Router(config)# vrf definition VRF_blue
Router(config-vrf)# rd 1:1
Router(config-vrf)# vpn id 1:1
Router(config-vrf)# address-family ipv4
Router(config-vrf-af)# mdt default mpls mldp 1.1.1.1
Router(config-vrf-af)# mdt data mpls mldp 100
Router(config-vrf-af)# mdt data threshold 4000000
Router(config-vrf-af)# route-target export 1:1
Router(config-vrf-af)# route-target import 1:1
Router(config-vrf-af)# exit
Router(config-vrf)# exit
Router(config)# ip multicast-routing vrf blue distributed
Router(config)# interface Loopback 0
Router(config-if)# ip address 1.1.1.1 255.255.255.0
Router(config-if)# exit
Router(config)# interface Loopback 1
Router(config-if)# vrf forwarding blue
Router(config-if)# ip address 192.0.100.1 255.255.255.0
Router(config-if)# ip pim sparse-mode
Router(config-if)# exit
```

```
Router(config)# interface GigabitEthernet 0/0/0
Router(config-if)# ip address 10.0.0.21 255.255.255.0
Router(config-if)# exit
Router(config)# interface Tunnel 100
Router(config-if)# ip address 10.0.0.1 255.255.255.0
Router(config-if)# mpls ip
Router(config-if)# tunnel source 10.0.0.21
Router(config-if)# tunnel destination 10.0.0.22
Router(config-if)# exit
Router(config-if)# end
```

The following example shows how to configure MVPNv4 MLDP over GRE on router PE2:

```
Router# enable
Router# configure terminal
Router(config)# vrf definition VRF_blue
Router(config-vrf)# rd 1:1
Router(config-vrf)# vpn id 1:1
Router(config-vrf)# address-family ipv4
Router(config-vrf-af)# mdt default mpls mldp 1.1.1.1
Router(config-vrf-af)# mdt data mpls mldp 100
Router(config-vrf-af)# mdt data threshold 1000
Router(config-vrf-af)# route-target export 1:1
Router(config-vrf-af)# route-target import 1:1
Router(config-vrf-af)# exit
Router(config-vrf)# exit
Router(config)# ip multicast-routing vrf blue distributed
Router(config)# interface Loopback 0
Router(config-if)# ip address 2.2.2.2 255.255.255.0
Router(config-if)# exit
Router(config)# interface Loopback 1
Router(config-if)# vrf forwarding blue
Router(config-if)# ip address 192.0.100.20 255.255.255.0
Router(config-if)# ip pim sparse-mode
Router(config-if)# exit
Router(config)# interface GigabitEthernet 0/0/0
Router(config-if)# ip address 10.0.0.22 255.255.255.0
Router(config-if)# exit
Router(config)# interface Tunnel 100
Router(config-if)# ip address 10.0.0.5 255.255.255.0
Router(config-if)# mpls ip
Router(config-if)# tunnel source 10.0.0.22
Router(config-if)# tunnel destination 10.0.0.21
Router(config-if)# exit
Router(config-if)# end
```

- To display the IPv6 neighbor information, use the **show ipv6 pim vrf** *vrf-name* **neighbor** command:

```
Router# show ipv6 pim vrf vrf blue neighbor

PIM Neighbor Table

Mode: B - Bidir Capable, G - GenID Capable

Neighbor Address           Interface          Uptime    Expires  Mode DR pri

::FFFF:1.1.1.1             Lspvif             3w0d      00:01:17 B G     1
```

Here, 1.1.1.1 is the loopback IP address of another PE on the other end of GRE tunnel, and ::FFFF:x.x.x.x is IPv4-mapped IPv6 IP address.

- To display the IPv4 neighbor information, use the **show ip pim vrf** *vrf-name* **neighbor** command:

```
Router# show ip pim vrf blue neighbor
PIM Neighbor Table
Mode: B - Bidir Capable, DR - Designated Router, N - Default DR Priority,
      P - Proxy Capable, S - State Refresh Capable, G - GenID Capable
Neighbor          Interface               Uptime/Expires    Ver   DR
Address                                                           Prio/Mode
30.2.0.3          Gi0/0/1.3900            2w0d/00:01:37     v2    0 / G
1.1.1.1           Lspvif                  7w0d/00:01:18     v2    1 / B S P G
```

- To display the IPv6 multicast routing table, use the **show ipv mroute vrf** *vrf-name* command:

```
Router# show ipv mroute vrf vrf blue

Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group,
       C - Connected, L - Local, I - Received Source Specific Host Report,
       P - Pruned, R - RP-bit set, F - Register flag, T - SPT-bit set,
       J - Join SPT, Y - Joined MDT-data group,
       y - Sending to MDT-data group
       g - BGP signal originated, G - BGP Signal received,
       N - BGP Shared-Tree Prune received, n - BGP C-Mroute suppressed,
       q - BGP Src-Active originated, Q - BGP Src-Active received
       E - Extranet
Timers: Uptime/Expires
Interface state: Interface, State

(2002:30::100, FF33:0:3::4000:1), 00:01:06/00:02:53, flags: sT
  Incoming interface: Lspvif1
  RPF nbr: ::FFFF:1.1.1.2
  Immediate Outgoing interface list:
    GigabitEthernet0/0/1.3900, Forward, 00:01:06/00:02:53
```

- To display the IPv4 multicast routing table, use the **show ip mroute** *vrf-name* command:

```
Router# show ip mroute vrf blue

IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
       L - Local, P - Pruned, R - RP-bit set, F - Register flag,
       T - SPT-bit set, J - Join SPT, M - MSDP created entry, E - Extranet,
       X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
       U - URD, I - Received Source Specific Host Report,
       Z - Multicast Tunnel, z - MDT-data group sender,
       Y - Joined MDT-data group, y - Sending to MDT-data group,
       G - Received BGP C-Mroute, g - Sent BGP C-Mroute,
       N - Received BGP Shared-Tree Prune, n - BGP C-Mroute suppressed,
       Q - Received BGP S-A Route, q - Sent BGP S-A Route,
       V - RD & Vector, v - Vector, p - PIM Joins on route,
       x - VxLAN group
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode

(30.0.0.100, 232.0.0.1), 1w0d/00:01:47, flags: sT
  Incoming interface: Null, RPF nbr 1.1.1.1
  Outgoing interface list:
    Gi0/0/1.3900, Forward/Sparse, 1w0d/00:01:47
```

- To display the multicast routing counter for IPv6, use the **show ipv6 mroute vrf** *vrf-name* **counter** command:

```
Router# show ipv6 mroute vrf vrf blue counter

Forwarding Counts: Pkt Count/Pkts per second/Avg Pkt Size/Kilobits per second
Other counts:      Total/RPF failed/Other drops(OIF-null, rate-limit etc)
VRF vrf blue
 5057 routes, 11 (*,G)s, 46 (*,G/m)s
Group: FF00::/8
  RP-tree,
   SW Forwarding: 0/0/0/0, Other: 0/0/0
   HW Forwarding:   NA/NA/NA/NA, Other: NA/NA/NA
Group: FF00::/15
  RP-tree,
   SW Forwarding: 0/0/0/0, Other: 0/0/0
   HW Forwarding:   NA/NA/NA/NA, Other: NA/NA/NA
Group: FF02::/16
  RP-tree,
   SW Forwarding: 0/0/0/0, Other: 3/3/0
Group: FF10::/15
  RP-tree,
   SW Forwarding: 0/0/0/0, Other: 0/0/0
   HW Forwarding:   NA/NA/NA/NA, Other: NA/NA/NA
Group: FF12::/16
  RP-tree,
   SW Forwarding: 0/0/0/0, Other: 0/0/0
Group: FF20::/15
  RP-tree,
   SW Forwarding: 0/0/0/0, Other: 0/0/0
   HW Forwarding:   NA/NA/NA/NA, Other: NA/NA/NA
Group: FF22::/16
  RP-tree,
   SW Forwarding: 0/0/0/0, Other: 0/0/0
Group: FF30::/15
  RP-tree,
   SW Forwarding: 0/0/0/0, Other: 0/0/0
   HW Forwarding:   NA/NA/NA/NA, Other: NA/NA/NA
Group: FF32::/16
  RP-tree,
   SW Forwarding: 0/0/0/0, Other: 0/0/0
Group: FF33::/32
  RP-tree,
   SW Forwarding: 0/0/0/0, Other: 0/0/0
   HW Forwarding:   NA/NA/NA/NA, Other: NA/NA/NA                ------- from the first
entry to this, all of these are default entries in IPv6 Mroute table
Group: FF33:0:3::4000:1
------- from this entry, all entries below are user entries learnt via PIM6 or MLD
protocol
  Source: 2002:30::100,
   SW Forwarding: 0/0/0/0, Other: 0/0/0
   HW Forwarding:   NA/NA/NA/NA, Other: NA/NA/NA
```

- To display the multicast routing counter for IPv4, use the **show ip mroute vrf** *vrf-name* **counter** command:

```
Router# show ip mroute vrf blue counter

Use "show ip mfib count" to get better response time for a large number of mroutes.

IP Multicast Statistics
5001 routes using 3706920 bytes of memory
101 groups, 49.50 average sources per group
Forwarding Counts: Pkt Count/Pkts per second/Avg Pkt Size/Kilobits per second
```

```
Other counts: Total/RPF failed/Other drops(OIF-null, rate-limit etc)

Group: 232.0.0.1, Source count: 50, Packets forwarded: 0, Packets received: 0
  Source: 30.0.0.149/32, Forwarding: 0/0/0/0, Other: 0/0/0
```

- To display the MPLS information, use the **show mpls forwarding-table labels** <*local label*> **detail** command:

```
Router# show mpls forwarding-table labels 10333 detail
Local      Outgoing   Prefix           Bytes Label   Outgoing   Next Hop
Label      Label      or Tunnel Id     Switched      interface
10333      No Label   [mdt 200:1 0][V] 0             aggregate/vrf-name
        MAC/Encaps=0/0, MRU=0, Label Stack{}, via Ls1
        VPN route: vrf blue
        No output feature configured
    Broadcast

Router# show mpls forwarding-table labels 1715 detail
Local      Outgoing   Prefix           Bytes Label   Outgoing   Next Hop
Label      Label      or Tunnel Id     Switched      interface
1715       No Label   [mdt 200:1 0][V] 0             aggregate/vpn200
        MAC/Encaps=0/0, MRU=0, Label Stack{}, via Ls1
        VPN route: vpn200
        No output feature configured
    Broadcast
```

- To display the MFIB table, use the **show mfib** <*vrf_name*> **verbose** command:

```
Router# show ip mfib vrf blue verbose
Entry Flags:    C - Directly Connected, S - Signal, IA - Inherit A flag,
                ET - Data Rate Exceeds Threshold, K - Keepalive
                DDE - Data Driven Event, HW - Hardware Installed
                ME - MoFRR ECMP entry, MNE - MoFRR Non-ECMP entry, MP - MFIB
                MoFRR Primary, RP - MRIB MoFRR Primary, P - MoFRR Primary
                MS  - MoFRR  Entry in Sync, MC - MoFRR entry in MoFRR Client.
I/O Item Flags: IC - Internal Copy, NP - Not platform switched,
                NS - Negate Signalling, SP - Signal Present,
                A - Accept, F - Forward, RA - MRIB Accept, RF - MRIB Forward,
                MA - MFIB Accept, A2 - Accept backup,
                RA2 - MRIB Accept backup, MA2 - MFIB Accept backup

Forwarding Counts: Pkt Count/Pkts per second/Avg Pkt Size/Kbits per second
Other counts:      Total/RPF failed/Other drops
I/O Item Counts:   FS Pkt Count/PS Pkt Count
VRF vpn200
 (*,224.0.0.0/4) Flags: K HW
   0x9A2  OIF-IC count: 0, OIF-A count: 0
   SW Forwarding: 0/0/0/0, Other: 0/0/0
   HW Forwarding:   NA/NA/NA/NA, Other: NA/NA/NA
 (*,224.0.1.40) Flags: C K HW
   0x9A4  OIF-IC count: 1, OIF-A count: 0
   SW Forwarding: 0/0/0/0, Other: 0/0/0
   HW Forwarding:   NA/NA/NA/NA, Other: NA/NA/NA
   Loopback200 Flags: RF F IC NS
     CEF: Special OCE (discard)
     Pkts: 0/0
 (*,232.0.0.0/8) Flags: K HW
   0x9A3  OIF-IC count: 0, OIF-A count: 0
   SW Forwarding: 0/0/0/0, Other: 0/0/0
   HW Forwarding:   NA/NA/NA/NA, Other: NA/NA/NA
 (30.0.0.100,232.0.0.1) Flags: K HW
   0x5C98  OIF-IC count: 0, OIF-A count: 0
   SW Forwarding: 0/0/0/0, Other: 0/0/0
   HW Forwarding:   NA/NA/NA/NA, Other: NA/NA/NA
```

```
GigabitEthernet0/0/1.3900 Flags: RF F NS
  CEF: Adjacency with MAC: 01005E000001503DE5974F0181000F3C0800
  Pkts: 0/0
```

Cisco ASR 1000 VxLAN Support

First Published: November 21, 2012

This module contains information about VxLAN (Virtual eXtensible Local Area Network) Layer 2 gateway feature support on the Cisco ASR 1000 Series Routers. VxLAN is a technology that provides a Layer 2 overlay network, allowing for network isolation. The standard 802.1q VLAN implementation limits the number of tags to 4096. However, cloud service providers may want to operate more than 4096 virtual networks. VxLAN uses a 24-bit network ID, which allows for a much larger number of individual i networks to be operated.

- Finding Feature Information
- Introduction
- Prerequisites for Cisco ASR 1000 Series Routers VxLAN Support
- Configuring the Cisco ASR 1000 Routers as a VxLAN Layer 2 Gateway with Multicast
- Configuring the Cisco ASR 1000 Routers as a VxLAN Layer 2 Gateway with Unicast
- Command Reference
- Additional References
- Feature Information for Cisco ASR 1000 Series Routers VxLAN Support

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Introduction

This feature enables the Cisco ASR 1000 Series Routers to act as a Layer 2 VxLAN gateway to provide support to bridge traffic across VxLAN segments in a hypervisor and on VLANs on physical servers. The operation of a VxLAN Layer 2 gateway is based on the data plane MAC address learning and flooding of multidestination traffic (such as unknown unicast, multicast, or broadcast frames) using IP multicast.

Acting as a VxLAN Layer 2 gateway, the Cisco ASR 1000 Routers can send and receive packets on multiple VxLAN networks, and provide connectivity between the hosts in a VLAN network and the virtual machines operating on a VxLAN network.

A VXLAN supports different modes for flood traffic:

- Multicast Mode—A VXLAN uses an IP multicast network to send broadcast, multicast, and unknown unicast flood frames. Each multicast mode VXLAN has an assigned multicast group IP address. When a new VM joins a host in a multicast mode VXLAN, a Virtual Ethernet Module (VEM) joins the assigned multicast group IP address by sending IGMP join messages. Flood traffic, broadcast, multicast and unknown unicast from the VM is encapsulated and is sent using the assigned multicast group IP address as the destination IP address. Packets sent to known unicast MAC addresses are encapsulated and sent directly to the destination server Virtual Tunnel Endpoint (VTEP) IP addresses.
- Unicast-Only Mode—A VXLAN uses each VEM's single unicast IP address as the destination IP address to send broadcast, multicast, and unknown unicast flood frames of the designated VTEP on each VEM that has at least one VM in the corresponding VXLAN. When a new VM joins the host in a unicast-mode VXLAN, a designated VTEP is selected for receiving flood traffic on that host. This designated VTEP is communicated to all other hosts through the Virtual Supervisor Module (VSM). Flood traffic (broadcast, multicast, and unknown unicast) is replicated on each VEM's designated VTEP in that VXLAN by encapsulating it with a VXLAN header. Packets are sent only to VEMs with a VM in that VXLAN. Packets that have a unicast MAC address are encapsulated and sent directly to the destination server's VTEP IP address.
- MAC Distribution Mode (supported only in unicast mode)—In this mode, unknown unicast flooding in the network is eliminated. The VSM learns all the MAC addresses from the VEMs in all the VXLANs and distributes those MAC addresses with VTEP IP mappings to other VEMs. Therefore, no unknown unicast MAC address exists in the network when the VMs on the VEMs are communicating and controlled by the same VSM.

The VxLAN Layer 2 gateway performs the following functions:

- Provides support to bridge traffic between a host in a VLAN domain and VMs behind a virtual switch (vSwitch) in a VxLAN domain. The VLAN and the virtual network identifier (VNI) on the VxLAN should be configured as member ports in the same bridge domain.
- Implements the Virtual Tunnel Endpoint (VTEP) function, which encapsulates the Layer 2 packet on the IP/UDP tunnel with the VxLAN header (VNI) information before sending it to a multicast group or particular virtual switch on the VxLAN domain.
- The VTEP function removes the VxLAN header, identifies the bridge domain under which the VNI is configured and then bridges the inner L2 packet to the VLAN side. The bridge function also learns the remote MAC address (the VM's MAC address behind the virtual switch).
- The Layer 2 gateway carries the inner payload of non-IP (Layer 2 traffic), IPv4, and IPv6 traffic over the VxLAN VNI member.

Prerequisites for Cisco ASR 1000 Series Routers VxLAN Support

The following are the prerequisites to configuring the Cisco ASR 1000 Routers as a VxLAN Layer 2 gateway:

1. Configure the loopback interface.
2. Configure the IP unicast reachability to remote VTEP's.
3. Configure Bidirectional Protocol Independent Multicast (PIM).

   For more information, see the *IP Multicast: PIM Configuration Guide, Cisco IOS XE Release 3S*.

Configuring the Cisco ASR 1000 Routers as a VxLAN Layer 2 Gateway with Multicast

- Configuring the VxLAN UDP Destination Port (Optional)
- Creating the Network Virtualization Endpoint (NVE) Interface
- Creating the Access Ethernet Flow Point (EFP)
- Mapping the VLAN to the Bridge Domain

Configuring the VxLAN UDP Destination Port (Optional)

The default VxLAN UDP destination is 8472. If you want to change the VxLAN UDP destination port value, you must change it before configuring the network virtualization endpoint (NVE) interface.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **vxlan udp port** *number*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `router# configure terminal`	Enters global configuration mode.
Step 3	**vxlan udp port** *number* **Example:** `Router(config)# vxlan udp port 1000`	Configures the VxLAN UDP destination port number. The default value is 8472.

Creating the Network Virtualization Endpoint (NVE) Interface

You create the network virtualization endpoint (NVE) interface and then assign member virtual network identifiers (VNIs) to it. The mapping between the VNI range and the multicast group range is either one-to-one or many-to-one.

SUMMARY STEPS

1. **interface nve** *number*
2. **source-interface loopback** *loopback interface*
3. **member vni** *range* **multicast-group** *range-group-address*
4. **no shutdown**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**interface nve** *number* **Example:** `Router(config)# interface nve 1`	Creates a network virtualization endpoint (NVE) interface and enters NVE interface configuration mode.
Step 2	**source-interface loopback** *number* **Example:** `Router(config-if)# source-interface loopback 0`	Assigns the previously-created loopback interface to the NVE interface.

	Command or Action	Purpose
Step 3	**member vni** {*range* | *startnumber-endnumber*} **multicast-group** *startip-address endip-address* **Example:** `Router(config-if)# member vni 7115 multicast-group 225.1.1.1`	Creates a VNI member or a range of VNI members. Repeat this step for each VNI to be added to the NVE interface. The valid values for the VNI number are from 4096 to 16777215.
Step 4	**no shutdown** **Example:** `Router(config-if)# no shutdown`	Enables the NVE interface.

Creating the Access Ethernet Flow Point (EFP)

After the member VNI is created, you must create the access Ethernet Flow Point (EFP) for the VLAN interface.

SUMMARY STEPS

1. **interface GigabitEthernet** *number*
2. **service instance** *id* **ethernet**
3. **encapsulation dot1q** *vlan-ID*
4. rewrite ingress tag pop 1 symmetric

DETAILED STEPS

	Command or Action	Purpose
Step 1	**interface GigabitEthernet** *number* **Example:** `Router(config)# interface GigabitEthernet1`	Enters interface configuration mode.
Step 2	service instance *id* ethernet **Example:** `Router(config-if)# service instance 20 ethernet`	Configures an Ethernet service instance on the overlay interface being configured and enters service instance configuration mode. • The service instance identifier range is from 1 to 8000.

	Command or Action	Purpose
Step 3	encapsulation dot1q *vlan-ID* **Example:** `Router(config-if-srv)# encapsulation dot1q 100`	Defines the VLAN encapsulation format as IEEE 802.1Q and specifies the VLAN identifier.
Step 4	rewrite ingress tag pop 1 symmetric **Example:** `Router(config-if-srv)# rewrite ingress tag pop 1 symmetric`	Removes the VLAN tag in the Layer 2 traffic before switching to the outgoing VxLAN interface. **Note** This command is required to remove the VLAN tag before sending the VLAN traffic to VxLAN and adding the VLAN tag in the reverse direction.

Mapping the VLAN to the Bridge Domain

You must map the VLAN created in the previous procedure to the bridge domain.

SUMMARY STEPS

1. **bridge-domain** *bridge-id*
2. **member** *interface* **service-instance** *id*
3. **member vni** *vni-id*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**bridge-domain** *bridge-id* **Example:** `Router(config)# bridge-domain 10`	Creates a bridge domain and enters bridge domain configuration mode. The valid range for bridge-id is 1-4096.
Step 2	**member** *interface* **service-instance** *id* **Example:** `Router(config-bdomain)# member gigabitEthernet 1 service-instance 1`	Binds the bridge domain to the service instance.
Step 3	**member vni** *vni-id* **Example:** `Router(config-bdomain)# member vni 1010`	Maps the VNI to the bridge domain.

The following example displays the NVE VNIs configured on the router:

```
Router# show nve vni
Interface  VNI          mcast       VNI state
nve1       1010         239.0.0.0      UP
nve2       2010         239.0.0.0      UP
```

The following example displays the NVE VNIs assigned to NVE interface 1:

```
Router(config)# show nve vni interface nve1
Interface  VNI          mcast       VNI state
nve1       1010        239.0.0.0      UP
nve1       1110        239.0.0.0      UP
```

The following example shows the status of NVE interface 1:

```
Router(config)# show nve interface nve1
Interface: nve1, State: Admin Up, Oper Up Encapsulation: Vxlan
source-interface: Loopback0 (primary:11.11.11.11 vrf:0)
```

The following example shows a detailed display for NVE interface 1:

```
Router(config)# show nve interface nve1 detail
Interface: nve1, State: Admin Up, Oper Up Encapsulation: Vxlan
source-interface: Loopback0 (primary:11.11.11.11 vrf:0)
Pkts In   Bytes In   Pkts Out  Bytes Out
0          0          0          0
```

The following example shows the NVE peers configured on the router:

```
Router(config)# show nve peers
Interface Peer-IP         VNI      Up Time
nve1       1.1.1.2        1010      10h
nve2       1.1.1.3        2030      20h
```

The following example shows the bridge domain configuration with the entry in bold displaying the VM's MAC address that was learned on the VxLAN VNI:

```
Router# show bridge-domain 1000
Bridge-domain 1000 (3 ports in all)
State: UP                    Mac learning: Enabled
Aging-Timer: 300 second(s)
    GigabitEthernet1 service instance 1000
    GigabitEthernet3 service instance 1000
    vni 7639335
   MAC address    Policy  Tag      Age  Pseudoport
   FFFF.FFFF.FFFF flood   static   0    OLIST_PTR:0xe9c2b410
   0050.56A4.ECD2 forward dynamic  297  nve1.VNI7639335 VxLAN [src:10.0.0.1  dst:10.0.0.2]
   0050.56A4.257A forward dynamic  297  GigabitEthernet3.EFP1000
```

Configuring the Cisco ASR 1000 Routers as a VxLAN Layer 2 Gateway with Unicast

The following example shows VxLAN with unicast ingress-replication which is a point-to-point (unicast) configuration.

```
interface Loopback0
ip address 11.11.11.11 255.255.255.255
!
interface nve1
no ip address
member vni 5001
  ingress-replication 22.22.22.22  < Remote L2 GW loopback ip>
!
source-interface Loopback0
!
```

```
bridge-domain 1
member vni 5001
member GigabitEthernet0/2/0 service-instance 1

interface GigabitEthernet0/2/0
service instance 1 ethernet
encapsulation dot1q 100
rewrite ingress tag pop 1 symmetric
```

Command Reference

- interface nve
- member
- member vni
- show nve interface
- show nve peers
- show nve vni
- source-interface loopback

interface nve

To create a network virtualization endpoint (NVE) interface and enter NVE interface configuration mode, use the **interface nve** command in global configuration mode. To remove the NVE interface, use the **no** form of this command.

interface nve *number*

no **interface nve** *number*

Syntax Description

number	The NVE interface number.

Command Modes

Global configuration (config)

Command History

Release	Modification
Cisco IOS XE Release 3.11S	This command was introduced on the Cisco ASR 1000 Series Routers.

Examples

```
The following example enters NVE interface mode:
Router(config)# interface nve 1
Router(config-if)#
```

Related Commands

Command	Description
source interface-loopback	Assigns a previously created loopback interface to an NVE interface.

member

To create a VNI member or range of members and map them to a multicast group, use the **member** command in NVE interface configuration mode. To delete the VNI member or range, use the **no** form of this command.

member vni {*number* | *startnumber-endnumber*} **multicast-group** *startip-address endip-address*

no member vni {*number* | *startnumber-endnumber*}

Syntax Description

vni	The member VNI.
number	The VNI number. The valid values are from 4096 to 16777215.
startnumber-endnumber	The starting and ending VNI numbers when entering a range.
multicast-group	The multicast group.
start-ip-address	The starting IPv4 address for the multicast group.
endip-address	The ending IPv4 address for the multicast group.

Command Modes

NVE interface configuration (config-if)

Command History

Release	Modification
Cisco IOS XE Release 3.11S	This command was introduced on the Cisco ASR 1000 Series Routers.

Examples

```
The following command creates VNI member 7115, and assigns it to NVE interface 1:
Router(config)# interface nve 1
Router(config-if)# member vni 7115 multicast-group 225.1.1.1

The following command creates a VNI member range from 6010 to 6030 and assigns it to NVE
interface 1:
Router(config)# interface nve 1
Router(config-if)# member vni 6010-6030 multicast-group 225.1.1.1
```

Related Commands

Command	Description
interface nve	Configures the NVE interface and enters NVE interface configuration mode.
source interface-loopback	Assigns a previously created loopback interface to an NVE interface.

member vni

To map a virtual network identifier to a bridge domain, use the **member vni** command in bridge-domain configuration mode. To remove the VNI from the bridge domain, use the **no** form of this command.

member vni *vni-id*

no member vni *vni-id*

Syntax Description

vni-id	The VNI number to be mapped to the bridge domain.

Command Modes

Bridge-domain configuration (config-bdomain)

Command History

Release	Modification
Cisco IOS XE Release 3.11S	This command was introduced on the Cisco ASR 1000 Series Routers.

Examples

```
The following example map a virtual network identifier to a bridge domain:
Router(config)# bridge-domain 10
Router(config-bdomain)# member vni 1010
```

Related Commands

Command	Description
member service-instance	Binds a bridge domain to the service instance.

show nve interface

To display the network virtualization endpoint (NVE) interface information, use the **show nve interface** command in privileged EXEC mode.

show nve interface *nve-interface* [**detail**]

Syntax Description

nve-interface	The NVE interface.
detail	(Optional) Displays detailed information about the NVE interface.

Command Modes

EXEC (#)

Command History

Release	Modification
Cisco IOS XE 3.11S	This command was introduced on the Cisco ASR 1000 Series Routers.

Examples

The following example shows the status of NVE interface 1:

```
Router# show nve interface nve1
Interface: nve1, State: Admin Up, Oper Up Encapsulation: Vxlan
source-interface: Loopback0 (primary:11.11.11.11 vrf:0)
```

The following example shows a detailed display for NVE interface 1:

```
Router# show nve interface nve1 detail
Interface: nve1, State: Admin Up, Oper Up Encapsulation: Vxlan
source-interface: Loopback0 (primary:11.11.11.11 vrf:0)
   Pkts In   Bytes In    Pkts Out  Bytes Out
         0          0          0          0
```

Table 19-2 describes the significant fields shown in the display.

Table 19-1 show nve interface Field Descriptions

Field	Description
VNI	The VNI number.
mcast	The multicast address.
VNI state	The current state of the VNI.

Related Commands

Command	Description
show nve peers	Displays the NVE peers configured on the router.
show nve vni	Displays the NVE VNIs configured on the router.

show nve peers

To display the network virtualization endpoint (NVE) peers configured on the router, use the **show nve peers** command in EXEC mode.

show nve peers

Syntax Description

This command has no arguments or keywords.

Command Modes

EXEC (#)

Command History

Release	Modification
Cisco IOS XE Release 3.11S	This command was introduced on the Cisco ASR 1000 Series Routers.

Examples

The following example shows the NVE peers configured on the router:

```
Router# show nve peers
Interface Peer-IP          VNI      Up Time
nve1       1.1.1.2         1010      10h
nve2       1.1.1.3         2030      20h
```

Table 19-2 describes the significant fields shown in the display.

Table 19-2 show nve peers Field Descriptions

Field	Description
Interface	The NVE interface.
Peer-IP	The peer IP address of the NVE interface.
VNI	The VNI number.
Up Time	The length of time for which the NVE interface has been up.

Related Commands

Command	Description
show nve interface	Displays the NVE interface information.
show nve vni	Displays the NVE VNIs configured on the router.

show nve vni

To display the network virtualization endpoint (NVE) VNIs configured on the router, use the **show nve vni** command in EXEC mode.

show nve vni [interface *nve-interface*]

Syntax Description

interface *nve-interface*	(Optional) Specifies the NVE interface.

Command Modes

EXEC (#)

Command History

Release	Modification
Cisco IOS XE Release 3.11S	This command was introduced on the Cisco ASR 1000 Series Routers.

Examples

The following example displays all the NVE VNIs configured on the router:

```
Router# show nve vni
Interface  VNI          mcast       VNI state
nve1       1010         239.0.0.0      UP
nve2       1110         239.0.0.0      UP
```

The following example displays the NVE VNIs assigned to NVE interface 1:

```
Router# show nve vni interface nve1
Interface  VNI          mcast       VNI state
nve1       1010         239.0.0.0       UP
nve1       1110         239.0.0.0       UP
```

Table 19-3 describes the significant fields shown in the display.

Table 19-3 show nve vni Field Descriptions

Field	Description
Interface	The NVE interface.
VNI	The VNI.
mcast	The multicast address for the NVE interface.
VNI state	The state of the VNI.

Related Commands

Command	Description
show nve interface	Displays the NVE interface information.
show nve peers	Displays the NVE peers configured on the router.

source-interface loopback

To assign a previously created loopback interface to an NVE interface, use the **source- interface loopbac**k command in NVE interface configuration mode. To remove the loopback interface from the NVE interface, use the **no** form of this command.

source-interface loopback *number*

no source-interface loopback *number*

Syntax Description

number	The loopback interface assigned to the NVE interface.

Command Modes

NVE interface configuration (config-if)

Command History

Release	Modification
Cisco IOS XE Release 3.11S	This command was introduced on the Cisco ASR 1000 Series Routers.

Examples

```
The following example maps a loopback interface to an NVE interface:
Router(config)# interface nve 1
Router(config-if)# source-interface loopback 0
```

Related Commands

Command	Description
interface nve	Configures the NVE interface and enters NVE interface configuration mode.

Additional References

Related Documents

Related Topic	Document Title

Feature Information for Cisco ASR 1000 Series Routers VxLAN Support

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 19-4 Feature Information for ASR 1000 Series Routers VxLAN Support

Feature Name	Releases	Feature Configuration Information
ASR 1000 Series Routers VxLAN Support	Cisco IOS XE Release 3.13.1S	This feature was introduced on the Cisco ASR 1000 Series Routers.

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Tracing and Trace Management

This chapter contains the following sections:

- Tracing Overview, page 20-1
- How Tracing Works, page 20-1
- Tracing Levels, page 20-2
- Viewing a Tracing Level, page 20-3
- Setting a Tracing Level, page 20-4
- Viewing the Content of the Trace Buffer, page 20-5

Tracing Overview

Tracing is a function that logs internal events. Trace files are automatically created and saved to the tracelogs directory on the harddisk: file system on all Cisco ASR 1000 Series Routers except the Cisco ASR 1002 Router, which stores tracing files in bootflash:. Trace files are used to store tracing data.

The contents of trace files are useful for the following purposes:

- Troubleshooting—If a Cisco ASR 1000 Series Router is having an issue, the trace file output may provide information that is useful for locating and solving the problem. Trace files can almost always be accessed through diagnostic mode even if other system issues are occurring.
- Debugging—The trace file outputs can help users get a more detailed view of system actions and operations.

How Tracing Works

The tracing function logs the contents of internal events on the Cisco ASR 1000 Series Routers. Trace files with all trace output for a module are periodically created and updated and are stored in the tracelog directory. Trace files can be erased from this directory to recover space on the file system without impacting system performance.

The most recent trace information for a specific module can be viewed using the **show platform software trace message** privileged EXEC and diagnostic mode command. This command can be entered to gather trace log information even during an IOS failure because it is available in diagnostic mode.

Trace files can be copied to other destinations using most file transfer functions (such as FTP, TFTP, and so on) and opened using a plaintext editor.

Tracing cannot be disabled on the Cisco ASR 1000 Series Router. Trace levels, however, which set the message types that generate trace output, are user-configurable and can be set using the **platform trace** commands. If a user wants to modify the trace level to increase or decrease the amount of trace message output, the user should set a new tracing level using the **platform trace** commands. Trace levels can be set by process using the **all-modules** keyword within the **platform trace** commands, or by module within a process. See the **platform trace** command reference for more information on this command, and the "Tracing Levels" section on page 20-2 of this document for additional information on tracing levels.

Tracing Levels

Tracing levels determine how much information about a module should be stored in the trace buffer or file.

shows all of the trace levels that are available and provides descriptions of what types of messages are displayed with each tracing level.

Table 20-1 Tracing Levels and Descriptions

Trace Level	Level Number	Description
Emergency	0	The message is regarding an issue that makes the system unusable.
Alert	1	The message is regarding an action that must be taken immediately.
Critical	2	The message is regarding a critical condition. This is the default setting for every module on the Cisco ASR 1000 Series Routers.
Error	3	The message is regarding a system error.
Warning	4	The message is regarding a system warning
Notice	5	The message is regarding a significant issue, but the router is still working normally.
Informational	6	The message is useful for informational purposes only.
Debug	7	The message provides debug-level output.
Verbose	8	All possible tracing messages are sent.
Noise	-	All possible trace messages for the module are logged. The noise level is always equal to the highest possible tracing level. Even if a future enhancement to tracing introduces a higher tracing level, the noise level will become equal to the level of that new enhancement.

Trace level settings are leveled, meaning that every setting will contain all messages from the lower setting plus the messages from its own setting. For instance, setting the trace level to 3(error) ensures that the trace file will contain all output for the 0 (emergencies), 1 (alerts), 2 (critical), and 3 (error) settings. Setting the trace level to 4 (warning) will ensure that all trace output for the specific module will be included in that trace file.

The default tracing level for every module on the Cisco ASR 1000 Series Router is notice.

All trace levels are not user-configurable. Specifically, the alert, critical, and notice tracing levels cannot be set by users. If you wish to trace these messages, set the trace level to a higher level that will collect these messages.

When setting trace levels, it is also important to remember that the setting is not done in a configuration mode, so trace level settings are returned to their defaults after every router reload.

Caution Setting tracing of a module to the debug level or higher can have a negative performance impact. Setting tracing to this level or higher should be done with discretion.

Caution Setting a large number of modules to high tracing levels can severely degrade performance. If a high level of tracing is needed in a specific context, it is almost always preferable to set a single module on a higher tracing level rather than setting multiple modules to high tracing levels.

Viewing a Tracing Level

By default, all modules on the Cisco ASR 1000 Series Routers are set to notice. This setting will be maintained unless changed by a user.

To see the tracing level for any module on the Cisco ASR 1000 Series Routers, enter the **show platform software trace level** command in privileged EXEC or diagnostic mode.

In the following example, the **show platform software trace level** command is used to view the tracing levels of the Forwarding Manager processes on the active RP:

```
Router# show platform software trace level forwarding-manager rp active
Module Name                      Trace Level
-------------------------------------------------
acl                              Notice
binos                            Notice
binos/brand                      Notice
bipc                             Notice
bsignal                          Notice
btrace                           Notice
cce                              Notice
cdllib                           Notice
cef                              Notice
chasfs                           Notice
chasutil                         Notice
erspan                           Notice
ess                              Notice
ether-channel                    Notice
evlib                            Notice
evutil                           Notice
file_alloc                       Notice
fman_rp                          Notice
fpm                              Notice
fw                               Notice
icmp                             Notice
interfaces                       Notice
iosd                             Notice
ipc                              Notice
ipclog                           Notice
iphc                             Notice
ipsec                            Notice
mgmte-acl                        Notice
mlp                              Notice
mqipc                            Notice
nat                              Notice
nbar                             Notice
netflow                          Notice
```

```
om                              Notice
peer                            Notice
qos                             Notice
route-map                       Notice
sbc                             Notice
services                        Notice
sw_wdog                         Notice
tdl_acl_config_type             Notice
tdl_acl_db_type                 Notice
tdl_cdlcore_message             Notice
tdl_cef_config_common_type      Notice
tdl_cef_config_type             Notice
tdl_dpidb_config_type           Notice
tdl_fman_rp_comm_type           Notice
tdl_fman_rp_message             Notice
tdl_fw_config_type              Notice
tdl_hapi_tdl_type               Notice
tdl_icmp_type                   Notice
tdl_ip_options_type             Notice
tdl_ipc_ack_type                Notice
tdl_ipsec_db_type               Notice
tdl_mcp_comm_type               Notice
tdl_mlp_config_type             Notice
tdl_mlp_db_type                 Notice
tdl_om_type                     Notice
tdl_ui_message                  Notice
tdl_ui_type                     Notice
tdl_urpf_config_type            Notice
tdllib                          Notice
trans_avl                       Notice
uihandler                       Notice
uipeer                          Notice
uistatus                        Notice
urpf                            Notice
vista                           Notice
wccp                            Notice
```

Setting a Tracing Level

To set a tracing level for any module on the Cisco ASR 1000 Series Routers, or for all modules within a process on the Cisco ASR 1000 Series Router, enter the **platform software trace** privileged EXEC and diagnostic mode command.

In the following example, the trace level for the forwarding processor module in the Forwarding Manager of the ESP processor in slot 0 is set to the informational tracing level (info):

```
Router(config)# platform trace runtime slot F0 bay 0 process forwarding-manager module
interfaces level info
```

In the following example, the trace level for the forwarding processor module in the Forwarding Manager of the ESP processor in slot R0 is set to the informational tracing level (max):

```
Router(config)# platform trace boottime slot R0 bay 1 process forwarding-manager
forwarding-manager level max
```

See the **platform trace boottime process forwarding-manager moduleinterfaces** and **platform trace runtime process forwarding-manager moduleinterfaces** command reference for additional information about the options for this command.

Viewing the Content of the Trace Buffer

To view the trace messages in the trace buffer or file, enter the **show platform software trace message** privileged EXEC and diagnostic mode command.

In the following example, the trace messages for the Host Manager process in Route Processor slot 0 are viewed using the **show platform software trace message** command:

```
Router# show platform software trace message host-manager R0
08/23 12:09:14.408 [uipeer]: (info): Looking for a ui_req msg
08/23 12:09:14.408 [uipeer]: (info): Start of request handling for con 0x100a61c8
08/23 12:09:14.399 [uipeer]: (info): Accepted connection for 14 as 0x100a61c8
08/23 12:09:14.399 [uipeer]: (info): Received new connection 0x100a61c8 on descriptor 14
08/23 12:09:14.398 [uipeer]: (info): Accepting command connection on listen fd 7
08/23 11:53:57.440 [uipeer]: (info): Going to send a status update to the shell manager in
slot 0
08/23 11:53:47.417 [uipeer]: (info): Going to send a status update to the shell manager in
slot 0
```

Packet Trace

First Published: November 21, 2013

The Packet-Trace feature provides a detailed understanding of how data packets are processed by the Cisco IOS XE platform, and thus helps customers to diagnose issues and troubleshoot them more efficiently. This module provides information about how to use the Packet-Trace feature.

Finding Feature Information

Your software release might not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the "Feature Information for Packet Trace" section on page 11.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Contents

This module contains the following sections:

- Information About Packet Trace, page 2
- Usage Guidelines for Configuring Packet Trace, page 3
- Configuring Packet Trace, page 4
- Displaying Packet-Trace Information, page 7
- Removing Packet-Trace Data, page 7
- Configuration Examples for Packet Trace, page 7
- Additional References, page 10
- Feature Information for Packet Trace, page 11

Information About Packet Trace

The Packet-Trace feature provides three levels of inspection for packets: accounting, summary, and path data. Each level provides a detailed view of packet processing at the cost of some packet processing capability. However, Packet Trace limits inspection to packets that match the debug platform condition statements, and is a viable option even under heavy-traffic situations in customer environments.

Table 21-1 explains the three levels of inspection provided by packet trace.

Table 21-1 Packet-Trace Level

Packet-Trace Level	Description
Accounting	Packet-Trace accounting provides a count of packets that enter and leave the network processor. Packet-Trace accounting is a lightweight performance activity, and runs continuously until it is disabled.
Summary	At the summary level of packet trace, data is collected for a finite number of packets. Packet-Trace summary tracks the input and output interfaces, the final packet state, and punt, drop, or inject packets, if any. Collecting summary data adds to additional performance compared to normal packet processing, and can help to isolate a troublesome interface.
Path data	The packet-trace path data level provides the greatest level of detail in packet trace. Data is collected for a finite number of packets. Packet-Trace path data captures data, including a conditional debugging ID that is useful to correlate with feature debugs, a timestamp, and also feature-specific path-trace data. Path data also has two optional capabilities: packet copy and Feature Invocation Array (FIA) trace. The packet-copy option enables you to copy input and output packets at various layers of the packet (layer 2, layer 3 or layer 4). The FIA- trace option tracks every feature entry invoked during packet processing and helps you to know what is happening during packet processing. **Note** Collecting path data consumes more packet-processing resources, and the optional capabilities incrementally affect packet performance. Therefore, path-data level should be used in limited capacity or in situations where packet performance change is acceptable.

Usage Guidelines for Configuring Packet Trace

Consider the following best practices while configuring the Packet-Trace feature:

- Use of ingress conditions when using the Packet-Trace feature is recommended for a more comprehensive view of packets.
- Packet-trace configuration requires data-plane memory. On systems where data-plane memory is constrained, carefully consider how you will select the packet-trace values. A close approximation of the amount of memory consumed by packet trace is provided by the following equation:

  memory required = (statistics overhead) + number of packets * (summary size + data size + packet copy size).

  When the Packet-Trace feature is enabled, a small, fixed amount of memory is allocated for statistics. Similarly, when per-packet data is captured, a small, fixed amount of memory is required for each packet for summary data. However, as shown by the equation, you can significantly influence the amount of memory consumed by the number of packets you select to trace, and whether you collect path data and copies of packets.

Configuring Packet Trace

Perform the following steps to configure the Packet-Trace feature.

Note The amount of memory consumed by the Packet-Trace feature is affected by the packet-trace configuration. You should carefully select the size of per-packet path data and copy buffers and the number of packets to be traced in order to avoid interrupting normal services. You can check the current data-plane DRAM memory consumption by using the **show platform hardware qfp active infrastructure exmem statistics** command. See the "Configuration Examples for Packet Trace" section on page 7 for more information about configuring packet trace.

SUMMARY STEPS

1. **enable**
2. **debug platform packet-trace enable**
3. **debug platform packet-trace packet** *pkt-num* **[fia-trace | summary-only] [data-size** *data-size*] **[circular]**
4. **debug platform packet-trace copy packet {input | output | both} [size** *num-bytes*] **[L2 | L3 | L4]**
5. **debug platform condition [ipv4 | ipv6] [interface** *interface*] **[access-list** *access-list -name*| *ipv4-address* **/** *subnet-mask* **|** *ipv6-address* **/** *subnet-mask*] **[ingress | egress]**
6. **debug platform packet-trace drop [code** *code-num*]
7. **debug platform condition start**
8. **debug platform condition stop**
9. **show platform packet-trace {configuration | statistics | summary | packet {all |** *pkt-num*}}
10. **clear platform condition all**
11. **exit**

DETAILED STEPS

	Command	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables the privileged EXEC mode. Enter your password if prompted.
Step 2	**debug platform packet-trace enable** **Example:** `Router# debug platform packet-trace enable`	Enables packet-trace accounting. **Note** This command is required for all levels of inspection.
Step 3	**debug platform packet-trace packet** *pkt-num* **[fia-trace | summary-only] [circular] [data-size** *data-size***]** **Example:** `Router# debug platform packet-trace packets 2048 summary-only`	Collects summary data for a specified number of packets. Captures feature path data by the default, and optionally performs FIA trace. *pkt-num*—Specifies the maximum number of packets maintained at a given time. **fia-trace**—Provides detailed level of data capture, including summary data, feature-specific data. Also displays each feature entry visited during packet processing. **summary-only**—Enables the capture of summary data with minimal details. **circular**—Saves the data of the most recently traced packets. *data-size*—Specifies the size of data buffers for storing feature and FIA trace data for each packet in bytes. When very heavy packet processing is performed on packets, users can increase the size of the data buffers if necessary. The default value is 2048.
Step 4	**debug platform packet-trace copy packet {in | out | both} [L2 | L3 | L4] [size** *num-bytes***]** **Example:** `Router# debug platform packet-trace copy packet input size 2048`	Enables copying of ingress or egress packets or both. Also, optionally, allows specifying the layer of the packet in which the packet copy should start. If the specified layer does not exist for the packet, the default behavior is to copy the packet at layer 2. *num-bytes*—Specifies the maximum number of octets to copy. Default value is 64.
Step 5	**debug platform condition [ipv4 | ipv6] [interface** *interface***][access-list** *access-list -name* **|** *ipv4-address* **/** *subnet-mask* **|** *ipv6-address* **/** *subnet-mask***] [ingress| egress]** **Example:** `Router# debug platform condition interface g0/0/0 ingress`	Specifies the matching criteria for tracing packets. Provides the ability to filter by protocol, IP address and subnet mask, access control list (ACL), interface, and direction.

	Command	Purpose
Step 6	**debug platform packet-trace drop [code** *code-num***]** **Example:** `Router# debug platform packet-trace drop`	Captures only the data for dropped packets, and provides the option to capture data for a specific global drop code. This command can be used with or without platform conditions. When this command is used without platform debug conditions, all the drop events in the system will be traced and packet trace will only capture the data available at the moment of the drop. When this command is used with platform debug conditions, packet-trace data is collected for all the packets classified as a match, but the collected data is preserved only for packets that meet the drop criteria. *code-num*—Specifies the drop code to trace.
Step 7	**debug platform condition start** **Example:** `Router# debug platform condition start`	Enables the specified matching criteria and starts packet tracing.
Step 8	**debug platform condition stop** **Example:** `Router# debug platform condition start`	Deactivates the condition and stops packet tracing.
Step 9	**show platform packet-trace {configuration | statistics | summary | packet {all |** *pkt-num***}}** **Example:** `Router# show platform packet-trace 14`	Displays packet-trace data according to the specified option. See Table 21-1 for detailed information about the **show** command options.
Step 10	**clear platform condition all** **Example:** `Router(config)# clear platform condition all`	Removes the configurations provided by the **debug platform condition** and **debug platform packet-trace** commands.
Step 11	**exit** **Example:** `Router# exit`	Exits the privileged EXEC mode.

Displaying Packet-Trace Information

Use these **show** commands to display packet-trace information.

Table 21-2 show Commands

Command	Description
show platform packet-trace configuration	Displays packet trace configuration, including any defaults.
show platform packet-trace statistics	Displays accounting data for all the traced packets.
show platform packet-trace summary	Displays summary data for the number of packets specified.
show platform packet-trace {all | *pkt-num*} **[decode]**	Displays the path data for all the packets or the packet specified. The **decode** option attempts to decode the binary packet into a more human- readable form.

Removing Packet-Trace Data

Use these commands to clear packet-trace data.

Table 21-3 clear Commands

Command	Description
clear platform packet-trace statistics	Clears the collected packet-trace data and statistics.
clear platform packet-trace configuration	Clears the packet-trace configuration and the statistics.

Configuration Examples for Packet Trace

This section provides the following configuration examples:

- Example: Configuring Packet Trace, page 7
- Example: Using Packet Trace, page 8

Example: Configuring Packet Trace

This example describes how to configure packet trace and display the results. In this example, incoming packets to Gigabit Ethernet interface 0/0/2 are traced, and FIA-trace data is captured for the first 128 packets. Also, the input packets are copied. The **show platform packet-trace packet 0** command displays the summary data and each feature entry visited during packet processing for packet 0.

```
Router> enable
Router# debug platform packet-trace enable
Router# debug platform packet-trace packet 128 fia-trace
Router# debug platform packet-trace copy packet input
Router# debug platform condition interface g0/0/2 ingress
Router# debug platform condition start
Router#! ping to UUT
Router# debug platform condition stop
Router# show platform packet-trace packet 0
Packet: 0           CBUG ID: 0
```

```
Summary
  Input     : GigabitEthernet0/0/0
  Output    : GigabitEthernet0/0/0
  State     : CONS Packet Consumed
  Timestamp : 260253459452456
Path Trace
  Feature: IPV4
    Source      : 1.1.1.2
    Destination : 1.1.1.1
    Protocol    : 1 (ICMP)
  Feature: FIA_TRACE
    Entry     : 0x8037eab4 - DEBUG_COND_INPUT_PKT
    Timestamp : 260253459469426
  Feature: FIA_TRACE
    Entry     : 0x82000d00 - IPV4_INPUT_DST_LOOKUP_CONSUME
    Timestamp : 260253459481199
  Feature: FIA_TRACE
    Entry     : 0x82000130 - IPV4_INPUT_FOR_US_MARTIAN
    Timestamp : 260253459495863
  Feature: FIA_TRACE
    Entry     : 0x80c782a4 - IPV4_INPUT_VFR
    Timestamp : 260253459624349
  Feature: FIA_TRACE
    Entry     : 0x82000e80 - IPV4_OUTPUT_LOOKUP_PROCESS
    Timestamp : 260253459633503
  Feature: ICMPV4
    Type : 0 (Echo Reply)
    Code : 0 (No Code)
  Feature: FIA_TRACE
    Entry     : 0x811f5028 - STILE_LEGACY_DROP
    Timestamp : 260253459841426
  Feature: FIA_TRACE
    Entry     : 0x808cfa1c - INPUT_FNF_AOR_DROP
    Timestamp : 260253459851553
  Feature: FIA_TRACE
    Entry     : 0x80824410 - INPUT_FNF_DROP
    Timestamp : 260253459859803
  Feature: FIA_TRACE
    Entry     : 0x808ce2a0 - INPUT_FNF_AOR_RELEASE
    Timestamp : 260253459872323
  Feature: FIA_TRACE
    Entry     : 0x802428ec - INPUT_DROP
    Timestamp : 260253459881419
Packet Copy In
  00235e49 7c000021 1bd1dc1a 08004500 00640005 0000ff01 b78f0101 01020101
  01010800 5faa0005 00000000 0000089f 15fcabcd abcdabcd abcdabcd abcdabcd
Router# clear platform condition all
Router# exit
```

Example: Using Packet Trace

This example provides a scenario in which packet trace is used to troubleshoot packet drops for a NAT configuration on a Cisco ASR 1006 Router. This example shows how you can effectively utilize the level of detail provided by the Packet-Trace feature to gather information about an issue, isolate the issue, and then find a solution.

In this scenario, you can detect that there are issues, but are not sure where to start troubleshooting. You should, therefore, consider accessing the Packet-Trace summary for a number of incoming packets.

```
Router# debug platform condition ingress
Router# debug platform packet-trace enable
```

```
Router# debug platform packet-trace packet 2048 summary-only
Router# debug platform condition start
Router# debug platform condition stop
Router# show platform packet-trace summary | include DROP
403     Gi0/0/0       ---           DROP   064.NatIn2Out
781     Gi0/0/0       ---           DROP   064.NatIn2Out
1708    Gi0/0/0       ---           DROP   064.NatIn2Out
```

The output shows that packets are dropped due to NAT configuration on Gigabit Ethernet interface 0/0/0, which enables you to understand that an issue is occurring on a specific interface. Using this information, you can limit which packets to trace, reduce the number of packets for data capture, and increase the level of inspection.

```
Router# debug platform packet-trace enable
Router# debug platform packet-trace packet 256
Router# debug platform packet-trace copy packet input size 32 L3
Router# debug platform condition interface Gi0/0/0
Router# debug platform condition start
Router# debug platform condition stop
Router# show platform packet-trace summary | include DROP
14      Gi0/0/0       ---           DROP   064.NatIn2Out
Router# show platform packet-trace 14
Packet: 14          CBUG ID: 17
Summary
  Input    : GigabitEthernet0/0/0
  Output   : ---
  State    : DROP    064.NatIn2Out
  Timestamp: 1123438869354000
Path Trace
  Feature: IPv4
    Source      : 10.1.1.1
    Destination : 64.102.6.247
    Protocol    : 17 (UDP)
  Feature: NAT
    Direction   : IN to OUT
    Action      : Drop
    Sub-code    : 027 - NON_PATTABLE
  Feature: DROP
    Code: 064.NatIn2Out
Packet Copy In
  45000040 00020000 3C11E030 0A010101 406606F7 046D0035
```

From the **show** command output, you can see that there is some traffic that cannot be configured for Port Address Translation (PAT). Hence, you should modify the ACL to spare this traffic from NAT configuration, or disable the service for that kind of traffic. From the copy of the incoming packet, you can see that the protocol is UDP, and the destination port is DNS, and that you should turn off the DNS Application Layer Gateway (ALG).

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	*Cisco IOS Master Commands List, All Releases*

Standards

Standard	Title
None	—

MIBs

MIB	MIBs Link
None	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at this URL: http://www.cisco.com/go/mibs

RFCs

RFC	Title
None	—

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for Packet Trace

Table 21-4 lists the features in this module and provides links to specific configuration information.

Use Cisco Feature Navigator to find information about platform support and software image support. Cisco Feature Navigator enables you to determine which software images support a specific software release, feature set, or platform. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Note Table 21-4 lists only the software releases that support a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Table 21-4 Feature Information for Packet Trace

Feature Name	Releases	Feature Information
Packet Trace	Cisco IOS XE 3.10	In Cisco IOS XE Release 3.10, the Packet-Trace feature was introduced on the Cisco ASR 1000 Series Router. The Packet-Trace feature provides information about how data packets are processed by the Cisco IOS XE platform. The following commands were introduced or modified: • **debug platform packet-trace enable** • **debug platform packet-trace packet** *pkt-num* **[fia-trace | summary-only] [data-size** *data-size***] [circular]** • **debug platform packet-trace copy packet {input | output | both} [size** *num-bytes***] [L2 | L3 | L4]** • **show platform packet-trace {configuration | statistics | summary | packet {all |** *pkt-num***}}**
	Cisco IOS XE 3.11	In Cisco IOS XE Release 3.11, the Packet-Trace feature was enhanced to include the following features: • Matched versus traced statistics. • Trace stop timestamp in addition to trace start timestamp. The following commands were introduced or modified: • **debug platform packet-trace drop [code** *drop-num***]** • **show platform packet-trace packet {all |** *pkt-num***} [decode]**

Configuring and Accessing the Web User Interface

The Cisco ASR 1000 Series Routers introduce a web user interface that expands on the functionality of web user interfaces available in other Cisco routers.

This chapter is about this web user interface and covers the following topics:

- Web User Interface Overview, page 22-1
- Configuring the Router for Web User Interface Access, page 22-5
- Authentication and the Web User Interface, page 22-7
- Domain Name System and the Web User Interface, page 22-7
- Clocks and the Web User Interface, page 22-7
- Accessing the Web User Interface, page 22-8
- Using Auto Refresh, page 22-9
- Web User Interface Tips and Tricks, page 22-10

Web User Interface Overview

This section covers the following topics:

- Web User Interface General Overview, page 22-1
- Legacy Web User Interface Overview, page 22-2
- Graphics-Based Web User Interface Overview, page 22-3
- Persistent Web User Interface Transport Maps Overview, page 22-4

Web User Interface General Overview

The web user interface is available on Cisco ASR 1000 Series Routers starting in Cisco IOS XE Release 2.1.1. The web user interface is not available in Cisco IOS XE Release 2.1.0.

The Cisco ASR 1000 Series Routers can be accessed using a web user interface. This web user interface allows users to monitor router performance using an easy-to-read graphical interface. Most aspects of a Cisco ASR 1000 Series Router can be monitored using the web user interface.

The web user interface has the following features:

- An interface that presents information in an easy-to-read graphical format.
- An interface that allows users to monitor most software processes, including processes related to the IOS and nonIOS subpackages within the Cisco IOS XE consolidated package.
- An interface that allows users to monitor most hardware components, including all RPs, ESPs, SIPs, and SPAs installed in the router.
- Access to the legacy web user interface in addition to the enhanced web user interface.
- The ability to gather the output of **show** commands from the web user interface.

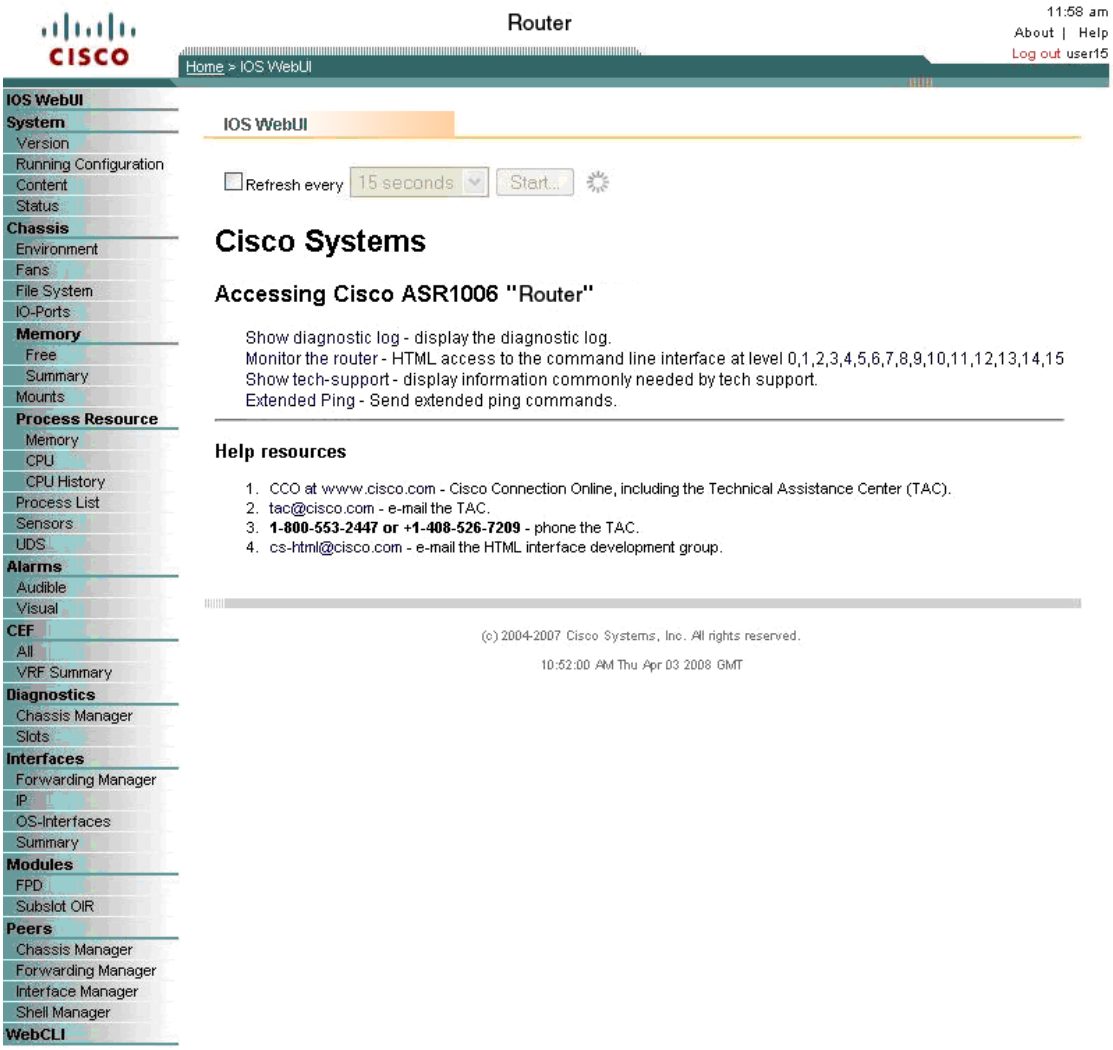
Legacy Web User Interface Overview

Previous Cisco routers have a legacy web user interface that can be used to monitor the router. This legacy web user interface presents information in a straightforward manner without using any graphics. On the Cisco ASR 1000 Series Routers, this interface is part of the larger web user interface and can be accessed by clicking the "IOS Web UI" option in the left-hand menu.

On the Cisco ASR 1000 Series Routers, the legacy web user interface can only be used to configure and monitor the IOS subpackage. In some scenarios, most notably when an **ip http** command has been successfully entered to enable the HTTP or HTTPS server while a properly configured web user interface transport map has not yet been applied on the Cisco ASR 1000 Series Router, the legacy web user interface will be accessible while the graphics-based web user interface will be inaccessible.

See Figure 22-1 for an example of the legacy web user interface home page.

Figure 22-1 Legacy Web User Interface Home Page



Graphics-Based Web User Interface Overview

The web user interface on the Cisco ASR 1000 Series Routers expands the legacy web user interface available on other platforms by presenting information in easy-to-read graphics-based tables, graphs, or charts, depending on the information presented. The web user interface on the Cisco ASR 1000 Series Routers is also able to present monitoring information stored in both the IOS and nonIOS subpackages, allowing for a complete view of the router using the web user interface.

See Figure 22-2 for an example of the graphics-based web user interface home page.

Figure 22-2 Graphics-Based Web User Interface Home Page

Persistent Web User Interface Transport Maps Overview

To enable the graphics-based web user interface, a persistent web user interface transport map must be configured. The persistent web user interface transport map, when successfully configured and applied to the router, defines how the router handles incoming web user interface requests. In the persistent web user interface transport map, users define whether the graphics-based web user interface can be accessed through HTTP, HTTPS, or both protocols. Only one persistent web user interface transport map can be applied to a Cisco ASR 1000 Series Router at a time.

The persistent web user interface transport map configuration must be performed in addition to the legacy web user interface configuration, which is configured using the **ip http** command set. The **ip http** command settings define which ports are used by HTTP or HTTPS for both the legacy and graphics-based web user interface.

For information on configuring the entire graphics-based web user interface, including the configuration of persistent web user interface transport maps, see the "Configuring the Router for Web User Interface Access" section on page 22-5.

Configuring the Router for Web User Interface Access

The ability to access either web user interface on the Cisco ASR 1000 Series Routers is disabled by default.

The legacy web user interface must be configured before the graphics-based web user interface can be enabled.

Note The web user interface will not work if the Management Ethernet interface has not been configured or is not working; specifically, the default route must be specified in the Management Ethernet VRF before the web user interface can be configured.
See the "" chapter for information on configuring the Management Ethernet interface on your router. See the "Setting a Default Route in the Management Ethernet Interface VRF" section on page 10-4 chapter for information on configuring a default route in the Management Ethernet interface on your router.

To enable the entire web user interface, perform the following tasks:

Step 1 (Optional) Ensure the clock setting on your router is accurate by entering the **show clock** command.

```
Router# show clock
*13:56:59.257 DST Mon May 5 2008
```

If the router time is not properly set, use the **clock set** and **clock timezone** commands for setting the router clock.

Note The "Clocks and the Web User Interface" section on page 22-7 provides additional information on how clock settings on both the router and the web-browser can impact the web user interface.

Step 2 Connect to your router and enter the **configure terminal** command to enter global configuration mode.

Step 3 Set the HTTP server authentication method to local by entering the **ip http authentication local** command.

Step 4 Enable the legacy web user interface by entering one of the following global configuration commands:

- **ip http server**—Enables HTTP on port 80, which is the default HTTP port.
- **ip http port** *port-number*—Enables HTTP on the nondefault user-specified port.
- **ip http secure-server**—Enables HTTPS on port 443, the default HTTPS port.
- **ip http secure-port** *port-number*—Enables HTTPS on the nondefault user-specified port.

The legacy web user interface becomes available at this point of the procedure. Users attempting to access the web user interface after this step is completed will see the legacy web user interface only.

To enable the graphics-based web user interface, proceed to Step 5 and complete the remaining steps in this procedure.

Step 5 Create and name a persistent web user interface transport map by entering the **transport-map type persistent webui** *transport-map-name* command.

Step 6 Enable HTTP, HTTPS, or both by entering the following commands in transport map configuration mode:

- **server**—Enables HTTP.
- **secure-server**—Enables HTTPS.

Port numbers cannot be set within the transport map. The port numbers defined in Step 4 are also used with these settings in the persistent web user interface transport map.

Step 7 (Optional) Enter the **show transport-map name** *transport-map-name* privileged EXEC command to verify that your transport map is properly configured.

Step 8 Enable the transport map by entering the **transport type persistent webui input** *transport-map-name* global configuration command.

Examples

In the following example, the HTTP server authentication method is set to local:

```
Router# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# ip http authentication local
Router(config)# exit
```

In the following example, the web user interface using the default HTTP port is enabled:

```
Router# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# ip http server
Router(config)# transport-map type persistent webui http-webui
Router(config-tmap)# server
Router(config-tmap)# exit
Router(config)# exit

Router# show transport-map name http-webui
Transport Map:
  Name: http-webui
  Type: Persistent Webui Transport

Webui:
  Server:        enabled
  Secure Server: disabled

Router# configure terminal
Router(config)# transport type persistent webui input http-webui
*Apr 22 02:43:55.798: %UICFGEXP-6-SERVER_NOTIFIED_START: R0/0: psd:  Server wui has been
notified to start
```

In the following example, the web user interface using the default HTTPs port is enabled:

```
Router# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# ip http secure-server
Router(config)# transport-map type persistent webui https-webui
Router(config-tmap)# secure-server
Router(config-tmap)# exit
Router(config)# transport type persistent webui input https-webui
```

```
*Apr 22 02:38:43.597: %UICFGEXP-6-SERVER_NOTIFIED_START: R0/0: psd:  Server wui has been
notified to start
```

In the following example, the web user interface using the default HTTP and HTTPS ports is enabled:

```
Router# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# ip http server
Router(config)# ip http secure-server
Router(config)# transport-map type persistent webui http-https-webui
Router(config-tmap)# server
Router(config-tmap)# secure-server
Router(config-tmap)# exit
Router(config)# transport type persistent webui input http-https-webui
*Apr 22 02:47:22.981: %UICFGEXP-6-SERVER_NOTIFIED_START: R0/0: psd:  Server wui has been
notified to start
```

Authentication and the Web User Interface

Users attempting to access the web user interface for a router are subject to the same authentication requirements configured for that router. The web browser prompts all users for a name and password combination, and the web browser then looks to the router configuration to see if a user should or should not be granted access to the web user interface.

Only users with a privilege level of 15 can access the web user interface. Otherwise, authentication of web user interface traffic is governed by the authentication configuration for all other traffic.

To configure authentication on your router, see *Configuring Authentication*.
http://www.cisco.com/en/US/docs/ios/12_2/security/configuration/guide/scfathen.html

Domain Name System and the Web User Interface

The Domain Name System (DNS) is a distributed database in which you can map hostnames to IP addresses through the DNS protocol from a DNS server.

If the router is configured to participate in the Domain Name System, users can access the web user interface by entering **http://***<dns-hostname>* as the web browser address.

For information on configuring DNS, see *Configuring DNS*.
http://www.cisco.com/en/US/docs/ios/ipaddr/configuration/guide/iad_config_dns_ps6922_TSD_Products_Configuration_Guide_Chapter.html

Clocks and the Web User Interface

Requests to view the web user interface can be rejected by certain web browsers if the time as seen by the web browser differs from the time as seen by the router by an hour or more.

For this reason, we recommend checking the router time using the **show clock** command before configuring the router and, if the router time is not properly set, use the **clock set** and **clock timezone** commands for setting the router clock.

Similarly, the web browser's clock source, which is usually the personal computer, must also have an accurate time to properly access the web user interface.

The following message appears when the web browser and the router clocks are more than an hour apart:

```
Your access is being denied for one of the following reasons:
. Your previous session has timed-out, or
. You have been logged out from elsewhere, or
. You have not yet logged in, or
. The resource requires a higher privilege level login.
```

If you see this message and fixing the other possible causes of the issue still does not make the web user interface accessible, check both the router clock and the PC clock to ensure both clocks reflect the accurate day and time and then retry your connection to the web user interface.

Also note that if one clock changes at daylight savings time while another clock does not, clock-related issues can occur.

Accessing the Web User Interface

To access the web user interface, perform the following tasks:

Step 1 Open your web browser. The web user interface supports the following web browsers:

- Microsoft Internet Explorer 6 or later
- Mozilla Firefox 2.0 or later

Step 2 Enter the address of the router in the address field of the web browser. The format for the address of the router in the address field is **http://**<*routername* or *management-ethernet-ip-address*>**:**[*http-port*] or **https://**<*routername* or *management-ethernet-ip-address*>**:**[*https-port*], and the addresses that are acceptable depend upon your web browser user interface configurations and whether your router is participating in DNS.
Following are some examples of acceptable address field web browser entries:

```
HTTP Using Default Port Example
http://172.16.5.1

HTTPS Using Default Port Example
https://172.16.5.1

HTTP Using NonDefault Port Example
http://172.16.5.1:94

HTTPS Using NonDefault Port Example
https://172.16.5.1:530/

HTTP Using Default Port Participating in DNS Example
http://router1

HTTPS Using Default Port Participating in DNS Example
https://router1

HTTP Using NonDefault Port Participating in DNS Example
http://router1:94

HTTPS Using NonDefault Port Participating in DNS Example
https://router1:530/
```

Step 3 If prompted, enter your username and password. The username and password combination required to enter the web user interface is the same combination required to access the router.

Step 4 The graphics-based web user interface similar to Figure 22-2 in the "Graphics-Based Web User Interface Overview" section on page 22-3 should appear in your web browser.

For additional information on the commands and the options available with each command, see the *Cisco IOS Configuration Fundamentals Command Reference Guide* located at: http://www.cisco.com/en/US/docs/ios/fundamentals/command/reference/cf_book.html.

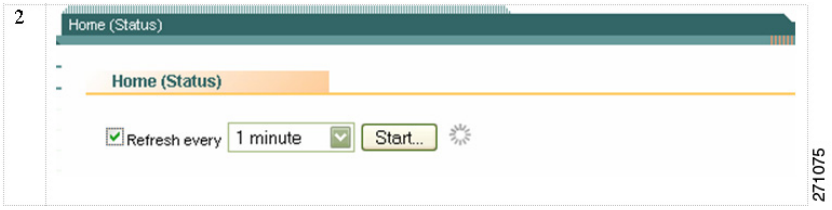
Using Auto Refresh

The web user interface does not refresh content automatically by default.

To set an auto-refresh interval, follow these steps:

Step 1 Check the **Refresh every** check box on the graphical web user interface home page. A check mark appears in the check box (See Figure 22-3).

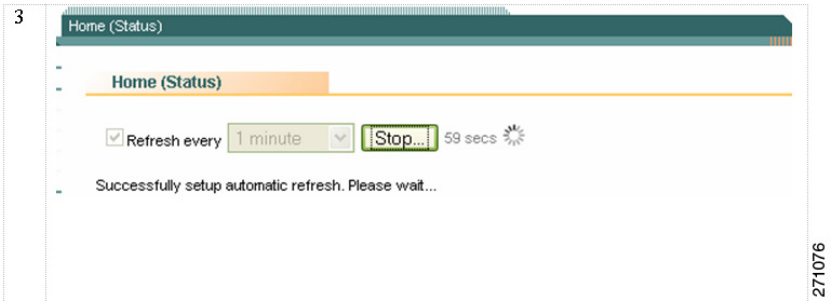
Figure 22-3 Checking Auto Refresh Check Box



Step 2 Set the frequency of the auto-refresh interval using the drop-down menu.

Step 3 Click the **Start** button to the right of the drop-down menu. After hitting this button, the **Start** button becomes the **Stop** button and a countdown timer placed to the right of the **Stop** button begins to increment (See Figure 22-4).

Figure 22-4 Auto Refresh Counter Example



The web user interface screen refreshes every time this counter reaches 0 seconds.

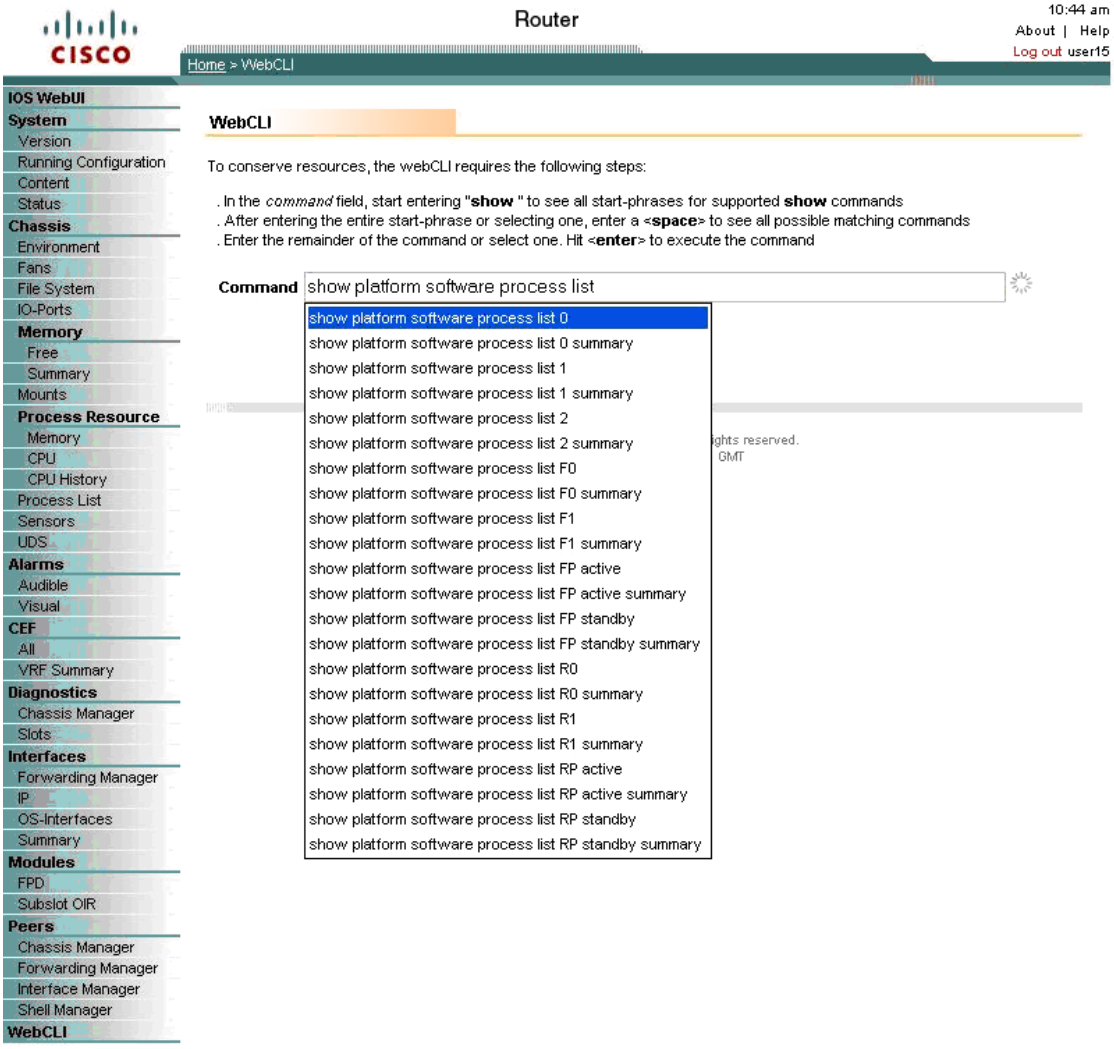
If you would like to stop the auto-refresh update, click the **Stop** button to return to the default setting of no auto-refresh update.

Web User Interface Tips and Tricks

This section provides some useful information about using the web user interface once the interface has been accessed:

- If you know a Cisco IOS command-line interface command to gather information that you are unable to gather using the web user interface, you can click **IOS Web UI** followed by **Monitor the Router** to enter commands.
- If you know a diagnostic mode command to gather information that you are unable to gather using the web user interface, you can click **WebCLI** to enter **show** commands.
- The WebCLI command line has a context-sensitive help feature that shows the options available in a certain keyword sequence using a drop-down menu.
  Figure 22-5 shows an example of this drop-down menu context-sensitive help feature.

Figure 22-5 Web CLI Drop-Down Menu



PPP Half-Bridge on the Cisco ASR 1000 Series Routers

First Published: December 12, 2014

For situations in which routed network needs connectivity to a remote bridged Ethernet network, a serial or ISDN interface can be configured to function as a PPP half-bridge. The line to the remote bridge functions as a virtual Ethernet interface, and the router's serial or ISDN interface functions as a node on the same Ethernet subnetwork as the remote network.

The PPP full-bridge sends PPP packets to the PPP half-bridge, which converts them to routed packets and forwards them to other router processes. Likewise, the PPP half-bridge converts routed packets to Ethernet bridge packets and sends them to the PPP full-bridge on the same Ethernet subnetwork.

Note An interface cannot function as both a half-bridge and a full-bridge.

Finding Feature Information

Your software release might not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the "Feature Information for PPP Half-Bridge on the Cisco ASR 1000 Series Routers" section on page 7.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Contents

This module contains the following sections:

- Prerequisites for PPP Half-Bridge on the Cisco ASR 1000 Series Routers, page 2
- Information About PPP Half-Bridge on the Cisco ASR 1000 Series Routers, page 2
- PPP Half-Bridge Supported Features, page 3
- Configuration Examples for PPP Half-Bridge, page 3

- Verifying PPP Half-Bridge, page 3
- Additional References, page 6
- Feature Information for PPP Half-Bridge on the Cisco ASR 1000 Series Routers, page 7

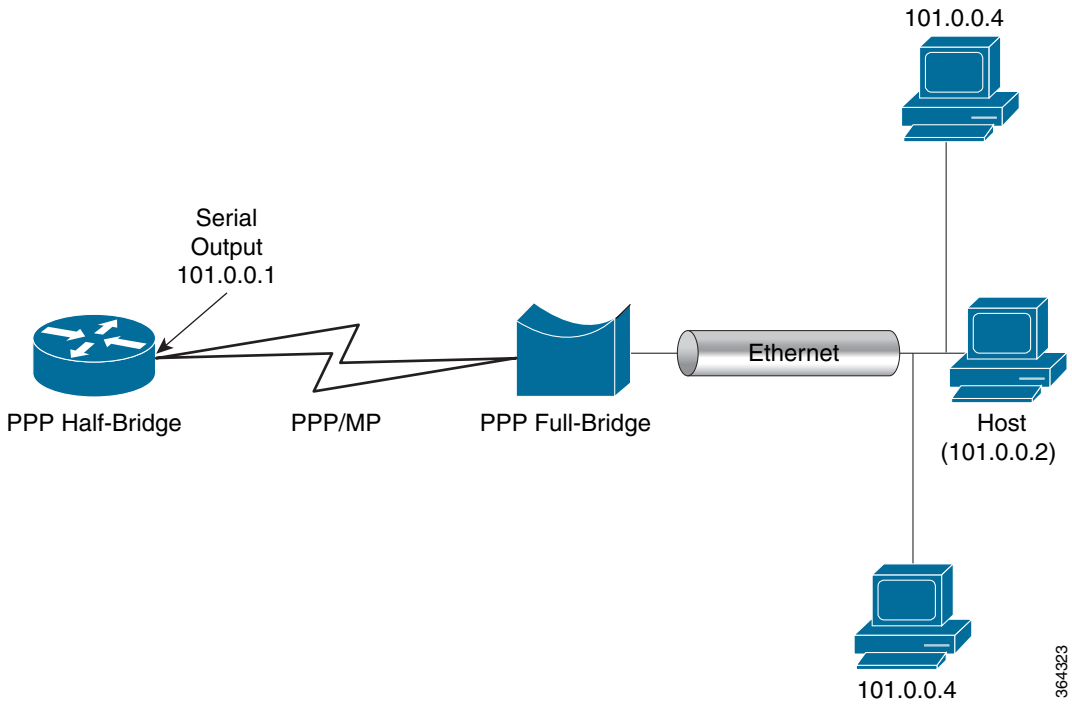
Prerequisites for PPP Half-Bridge on the Cisco ASR 1000 Series Routers

PPP encapsulation must be enabled under the interface to use PPP half-bridge on ASR 1000 Series Routers.

Information About PPP Half-Bridge on the Cisco ASR 1000 Series Routers

Figure 1 shows a half-bridge configuration. The router port connecting to the bridge is configured as a serial interface to the PPP half-bridge. The interface functions as a virtual node on the Ethernet subnetwork on the bridge. The serial interface has an IP address associated with the Ethernet subnetwork.

Figure 23-1 PPP Half-Bridge Configuration



When a packet destined to the Ethernet arrives at the PPP half-bridge, it is converted to Ethernet packets, encapsulated in PPP frames, and sent on the Ethernet bridge link. The PPP full-bridge decapsulates PPP header, and forwards the Ethernet packet with bridge functionality.

In the reverse direction, PPP half-bridge extracts the Ethernet packets encapsulated in PPP frames that are received from PPP full-bridge device. The PPP half-bridge converts them to IP packets, and routes them on the Internet.

PPP Half-Bridge Supported Features

PPP half-bridge supports the following interworking features:

- VRF aware PPP half bridge interface
- Multicast and VPN Multicast (IGMP V2 and V3) on the PPP half bridge interface
- Routing protocol on the PPP half bridge interface
- IPv4 on the PPP half bridge interface
- ACL on the PPP half bridge interface
- QOS on the PPP half bridge interface
- PPP authentication on the half bridge interface

Configuration Examples for PPP Half-Bridge

Example: Configuring PPP Half-Bridge on the Cisco ASR 1000 Series Routers

The following example shows how to configure PPP Half-Bridge on ASR 1000 Series Routers:

```
interface Serial0/1/1:0
 ip address 101.0.0.1 255.255.255.0
 encapsulation ppp
 ppp bridge ip
```

Example: Configuring PPP Full-Bridge on the Cisco 7206 Routers

The following example shows how to emulate the bridge device.

```
no ip routing
interface Serial5/1:0
 no ip address
 encapsulation ppp
 bridge-group 1
interface GigabitEthernet0/2
bridge-group 1
```

Verifying PPP Half-Bridge

The commands outlined in this section can be used to verify the PPP half-bridge.

Command	Purpose
show ppp interfaces	Displays PPP state information for an interface.
show adjacency	Displays information about the Cisco Express Forwarding adjacency table or the hardware Layer 3-switching adjacency table.
show interfaces serial slot/port	Displays information about a serial interface.

The following example displays PPP state information for an interface.

```
Router# show ppp all
Interface/ID OPEN+ Nego* Fail-     Stage    Peer Address    Peer Name
------------ -------------------- -------- --------------- --------------------
Se0/1/1:0    LCP+ BCP+            LocalT   0.0.0.0
```

The following example shows the sample output for **show adjacency** command.

```
Router# show adjacency serial 0/1/1:0 detail
Protocol Interface                 Address
IP       Serial0/1/1:0             101.0.0.2(11)
                                   0 packets, 0 bytes
                                   epoch 0
                                   sourced in sev-epoch 439
                                   Encap length 20
                                   FF0300310001503DE53411110022BDD5
                                   E8000800
                                   ARP
```

The following example displays information about a serial interface.

```
Router# show interfaces serial 0/1/1:0
Serial0/1/1:0 is up, line protocol is up
  Hardware is SPA-8XCHT1/E1
  Internet address is 101.0.0.1/24
  MTU 1500 bytes, BW 1984 Kbit/sec, DLY 20000 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation PPP, LCP Open
  Open: BCP, crc 16, loopback not set
  Keepalive set (10 sec)
  ARP type: ARPA, ARP Timeout 04:00:00
```

Debugging PPP Half-Bridge

The following commands are used to debug PPP half-bridge.

Command	Purpose
debug ppp negotiation	Enables debugging of LCP and Network Control Protocol (NCP) session negotiations.
debug ppp error	Displays local errors.

Command	Purpose
debug ppp events	Displays information about protocol actions and transitions between action states (pending, waiting, idle) on the link.
debug arp	Displays information on Address Resolution Protocol (ARP) transactions.

Sample output for debug ppp command

```
Router# debug ppp

Aug  6 14:36:00.021 CST: ppp827 PPP: Phase is ESTABLISHING
*Aug  6 14:36:00.021 CST: Se0/3/0:0 PPP: Using default call direction
*Aug  6 14:36:00.021 CST: Se0/3/0:0 PPP: Treating connection as a dedicated line*Aug  6
14:36:00.021 CST: Se0/3/0:0 PPP: Session handle[1900033B] Session id[827]
*Aug  6 14:36:00.021 CST: Se0/3/0:0 LCP: Event[OPEN] State[Initial to Starting]
*Aug  6 14:36:00.021 CST: Se0/3/0:0 LCP: O CONFREQ [Starting] id 1 len 10
*Aug  6 14:36:00.021 CST: Se0/3/0:0 LCP:    MagicNumber 0x6B5AC045 (0x05066B5AC045)*Aug  6
14:36:00.021 CST: Se0/3/0:0 LCP: Event[UP] State[Starting to REQsent]*
```

Sample output for debug arp command

```
Router# debug arp
Aug  6 15:43:22.915 CST: IP ARP: rcvd req src 101.0.0.2 503d.e534.1111, dst 101.0.0.1
Serial0/1/1:0
Aug  6 15:43:22.917 CST: IP ARP: creating entry for IP address: 101.0.0.2, hw:
503d.e534.1111
Aug  6 15:43:22.917 CST: IP ARP: sent rep src 101.0.0.1 0022.bdd5.e800, dst 101.0.0.2
503d.e534.1111 Serial0/1/1:2
```

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	*Cisco IOS Master Commands List, All Releases*

Standards

Standard	Title
None	—

MIBs

MIB	MIBs Link
None	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at this URL: http://www.cisco.com/go/mibs

RFCs

RFC	Title
None	—

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for PPP Half-Bridge on the Cisco ASR 1000 Series Routers

Table 23-1 lists the features in this module and provides links to specific configuration information.

Use Cisco Feature Navigator to find information about platform support and software image support. Cisco Feature Navigator enables you to determine which software images support a specific software release, feature set, or platform. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Note Table 23-1 lists only the software releases that support a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Table 23-1 Feature Information for PPP Half-Bridge on the Cisco ASR 1000 Series Routers

Feature Name	Releases	Feature Information
PPP Half-Bridge on the Cisco ASR 1000 Series Routers	Cisco IOS XE 3.13.1	In Cisco IOS XE Release 3.13.1, the PPP Half-Bridge feature was introduced on the Cisco ASR 1000 Series Router.

Unsupported Commands

The Cisco ASR 1000 Series Routers contain a series of commands with the **platform** keyword that either produce no output or produce output that is not useful for customer purposes. Although some commands with the **platform** keyword produce output that is useful for customer purposes, and are therefore documented in the Cisco IOS XE command references, the vast majority of these commands are not useful for customer purposes and are therefore considered unsupported.

Because these commands are considered unsupported, you will not find any further Cisco documentation for any of these commands.

Following is the list of unsupported commands for the Cisco ASR 1000 Series Routers:

- **clear platform hardware cpp**
- **clear platform software access-list**
- **clear platform software adjacency**
- **clear platform software cce**
- **clear platform software cef**
- **clear platform software cman**
- **clear platform software dpidb**
- **clear platform software ess**
- **clear platform software icmp**
- **clear platform software infrastructure**
- **clear platform software inspect**
- **clear platform software mlp**
- **clear platform software mpls**
- **clear platform software multicast**
- **clear platform software nat**
- **clear platform software nbar**
- **clear platform software netflow**
- **clear platform software pam**
- **clear platform software qos**
- **clear platform software route-map**
- **clear platform software shell**

- **clear platform software wccp**
- **clear platform software zone**
- **clear platform software zone-pair**
- **debug platform all**
- **debug platform hardware cpp**
- **debug platform software acl**
- **debug platform software adjacency**
- **debug platform software cce**
- **debug platform software cef**
- **debug platform software cman**
- **debug platform software configuration**
- **debug platform software dpidb**
- **debug platform software entity**
- **debug platform software ess**
- **debug platform software FPM**
- **debug platform software icmp**
- **debug platform software infrastructure**
- **debug platform software inspect**
- **debug platform software iphc**
- **debug platform software ipsec**
- **debug platform software issu**
- **debug platform software kernel**
- **debug platform software liin**
- **debug platform software marmot**
- **debug platform software mlp**
- **debug platform software mpls**
- **debug platform software multicast**
- **debug platform software nat**
- **debug platform software nbar**
- **debug platform software netflow**
- **debug platform software pam**
- **debug platform software qos**
- **debug platform software route-map**
- **debug platform software shell**
- **debug platform software urpf**
- **debug platform software vty**
- **debug platform software wccp**
- **debug platform software zone**

- **set platform software trace**
- **show crypto engine**
- **show platform hardware cpp (all commands)**
- **show platform software access-list**
- **show platform software adjacency**
- **show platform software bound-uds**
- **show platform software cce statistics**
- **show platform software cef counters**
- **show platform software cman statistics**
- **show platform software common-classification**
- **show platform software debug**
- **show platform software diagnostic**
- **show platform software dpidb**
- **show platform software erspan**
- **show platform software ess**
- **show platform software firewall**
- **show platform software flow-control identifier**
- **show platform software fpm**
- **show platform software icmp**
- **show platform software icmp6**
- **show platform software infrastructure**
- **show platform software inspect statistics**
- **show platform software interface**
- **show platform software ip**
- **show platform software ipc**
- **show platform software iphc**
- **show platform software ipsec**
- **show platform software ipv6**
- **show platform software loadinfo**
- **show platform software mlist**
- **show platform software mlp**
- **show platform software mlppp**
- **show platform software module**
- **show platform software mpls**
- **show platform software multicast**
- **show platform software nat**
- **show platform software nbar**
- **show platform software netflow**

- **show platform software object-manager**
- **show platform software os-interface**
- **show platform software pam**
- **show platform software peer**
- **show platform software punt**
- **show platform software push-counter**
- **show platform software qos**
- **show platform software sbc tables**
- **show platform software shell**
- **show platform software slot**
- **show platform software spi4**
- **show platform software urpf**
- **show platform software vlans**
- **show platform software vty**
- **show platform software wccp**
- **show platform software zone statistics**
- **test platform hardware eobc**
- **test platform hardware port**
- **test platform hardware slot**
- **test platform hardware subslot**
- **test platform software cef**
- **test platform software chassis**
- **test platform software debug**
- **test platform software forwarding-manager**
- **test platform software infrastructure**
- **test platform software inject**
- **test platform software process**
- **test platform software shell**
- **test platform software tdl**
- **test platform software trace**

Configuration Examples

This appendix is designed to provide examples of how some common networking tasks can be accomplished on the Cisco ASR 1000 Series Routers. The examples in this appendix are provided for illustrative purposes only; little or no context is given with these examples. See the other chapters in this book (in particular Chapter 3, "Consolidated Packages and SubPackages Management"), for more detailed explanations of the examples in this section.

When reading this section, also be aware that networking configurations are complex and can be configured many ways. The examples in this section simply show one method of accomplishing a configuration.

This appendix contains the following examples:

- Configuring the Router to Boot the Consolidated Package on the TFTP Server, page B-1
- Copying the Consolidated Package from the TFTP Server to the Router, page B-5
- Configuring the Router to Boot Using the Consolidated Package Stored on the Router, page B-6
- Extracting the SubPackages from a Consolidated Package into the Same File System, page B-7
- Extracting the SubPackages from a Consolidated Package into a Different File System, page B-9
- Configuring the Router to Boot Using the SubPackages, page B-10
- Backing Up Configuration Files, page B-14
- Enabling a Second IOS Process on a Single RP Using SSO, page B-15
- ISSU—Consolidated Package Upgrade, page B-19

Configuring the Router to Boot the Consolidated Package on the TFTP Server

The following example shows how to configure the router to boot the consolidated package on the TFTP server:

```
Router(config)# boot system
tftp://172.17.16.81/auto/tftp-users/user/asr1000rp1-adventerprisek9.02.01.00.122-33.XNA.bi
n
Router(config)#config-reg 0x2102
Router(config)# exit
*May 29 21:51:11.963: %SYS-5-CONFIG_I: Configured from console by user on console
Router# show run | include boot
boot-start-marker
```

```
boot system
tftp://172.17.16.81/auto/tftp-users/user/asr1000rp1-adventerprisek9.02.01.00.122-33.XNA.bi
n
boot-end-marker
Router# copy run start
Destination filename [startup-config]?
Building configuration...
[OK]
Router# reload
Proceed with reload? [confirm]

*May 29 21:52:01.500: %SYS-5-RELOAD: Reload requested by user on console. Reload Reason:
Reload command.


System Bootstrap, Version 12.2(33r)XN2, RELEASE SOFTWARE (fc1)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 2008 by cisco Systems, Inc.


Current image running: Boot ROM0
Last reset cause: LocalSoft

ASR1000-RP1 platform with 4194303 Kbytes of main memory


          IP_ADDRESS: 172.17.52.155
      IP_SUBNET_MASK: 255.255.255.0
     DEFAULT_GATEWAY: 172.17.52.1
         TFTP_SERVER: 172.17.16.81
           TFTP_FILE:
auto/tftp-users/user/asr1000rp1-adventerprisek9.02.01.00.122-33.XNA.bin
Using midplane macaddr
        TFTP_MACADDR: 00:1a:30:46:e3:ff
        TFTP_VERBOSE: Progress
    TFTP_RETRY_COUNT: 18
        TFTP_TIMEOUT: 7200
       TFTP_CHECKSUM: Yes
          ETHER_PORT: 3
    ETHER_SPEED_MODE: Auto Detect
link up 100Mbps/HD
Receiving auto/tftp-users/user/asr1000rp1-adventerprisek9.02.01.00.122-33.XNA.bin from
172.17.16.81
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!.!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!.
!!!!!!!!!!!!!!!!!!!!!!.!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!.!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
File reception completed.
Boot image size = 208904396 (0xc73a0cc) bytes

Using midplane macaddr
Package header rev 0 structure detected
```

```
Calculating SHA-1 hash...done
validate_package: SHA-1 hash:
        calculated fb456e80:4ba2fa07:8556d27c:ea643e4f:512236df
        expected   fb456e80:4ba2fa07:8556d27c:ea643e4f:512236df
Image validated


PPC/IOS XE loader version: 0.0.3
loaded at:     00800000 0CF3C004
zimage at:     00807673 009B92D6
initrd at:     009BA000 01041CC9
isord at:      01042000 0CF39800
avail ram:     00400000 00800000

Kernel load:
Uncompressing image... dst: 00000000 lim: 00400000 start: 00807673 size: 001B1C63...done.
Now booting the IOS XE kernel
%IOSXEBOOT-4-BOOT_SRC: (rp/0): Non-HD Boot
%IOSXEBOOT-4-BOOT_PARAMETER: (rp/0): Booting with custom BOOT_PARAM setting

              Restricted Rights Legend

Use, duplication, or disclosure by the Government is
subject to restrictions as set forth in subparagraph
(c) of the Commercial Computer Software - Restricted
Rights clause at FAR sec. 52.227-19 and subparagraph
(c) (1) (ii) of the Rights in Technical Data and Computer
Software clause at DFARS sec. 252.227-7013.

           cisco Systems, Inc.
           170 West Tasman Drive
           San Jose, California 95134-1706



Cisco IOS Software, IOS-XE Software (PPC_LINUX_IOSD-ADVENTERPRISEK9-M), Version
12.2(33)XNA, RELEASE SOFTWARE (fc1)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2008 by Cisco Systems, Inc.
Compiled Thu 01-May-08 00:29 by mcpre
Image text-base: 0x10064AF0, data-base: 0x137E0958



Cisco IOS-XE software, Copyright (c) 1986-2008 by Cisco Systems, Inc.
All rights reserved.  Certain components of Cisco IOS-XE software are
licensed under the GNU General Public License ("GPL") Version 2.0.  The
software code licensed under GPL Version 2.0 is free software that comes
with ABSOLUTELY NO WARRANTY.  You can redistribute and/or modify such
GPL code under the terms of GPL Version 2.0.  For more details, see the
documentation or "License Notice" file accompanying the IOS-XE software,
or the applicable URL provided on the flyer accompanying the IOS-XE
software.



This product contains cryptographic features and is subject to United
States and local country laws governing import, export, transfer and
use. Delivery of Cisco cryptographic products does not imply
third-party authority to import, export, distribute or use encryption.
Importers, exporters, distributors and users are responsible for
compliance with U.S. and local country laws. By using this product you
agree to comply with applicable laws and regulations. If you are unable
to comply with U.S. and local laws, return this product immediately.
```

```
A summary of U.S. laws governing Cisco cryptographic products may be found at:
http://www.cisco.com/wwl/export/crypto/tool/stqrg.html

If you require further assistance please contact us by sending email to
export@cisco.com.

cisco ASR1006 (RP1) processor with 1772131K/6147K bytes of memory.
5 Gigabit Ethernet interfaces
2 Packet over SONET interfaces
2 Channelized T3 ports
32768K bytes of non-volatile configuration memory.
4194304K bytes of physical memory.
921599K bytes of eUSB flash at bootflash:.
39004543K bytes of SATA hard disk at harddisk:.
Duplex is configured.Remove duplex configuration before enabling auto-negotiation

<output removed for brevity>

User Access Verification

Username: user
Password:

Router>en
Password:
Router# show version
Cisco IOS Software, IOS-XE Software (PPC_LINUX_IOSD-ADVENTERPRISEK9-M), Version
12.2(33)XNA, RELEASE SOFTWARE (fc1)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2008 by Cisco Systems, Inc.
Compiled Thu 01-May-08 00:29 by mcpre


Cisco IOS-XE software, Copyright (c) 1986-2008 by Cisco Systems, Inc.
All rights reserved.  Certain components of Cisco IOS-XE software are
licensed under the GNU General Public License ("GPL") Version 2.0.  The
software code licensed under GPL Version 2.0 is free software that comes
with ABSOLUTELY NO WARRANTY.  You can redistribute and/or modify such
GPL code under the terms of GPL Version 2.0.  For more details, see the
documentation or "License Notice" file accompanying the IOS-XE software,
or the applicable URL provided on the flyer accompanying the IOS-XE
software.


ROM: IOS-XE ROMMON

 Router uptime is 2 minutes
Uptime for this control processor is 4 minutes
System returned to ROM by reload at 14:52:01 DST Thu May 29 2008
System image file is "tftp://172.17.16.81/auto/tftp-users/user/asr1000rp1-adve"
Last reload reason: Reload command



This product contains cryptographic features and is subject to United
States and local country laws governing import, export, transfer and
use. Delivery of Cisco cryptographic products does not imply
third-party authority to import, export, distribute or use encryption.
Importers, exporters, distributors and users are responsible for
compliance with U.S. and local country laws. By using this product you
agree to comply with applicable laws and regulations. If you are unable
to comply with U.S. and local laws, return this product immediately.
```

```
A summary of U.S. laws governing Cisco cryptographic products may be found at:
http://www.cisco.com/wwl/export/crypto/tool/stqrg.html

If you require further assistance please contact us by sending email to
export@cisco.com.

cisco ASR1006 (RP1) processor with 1772131K/6147K bytes of memory.
5 Gigabit Ethernet interfaces
8 Serial interfaces
8 Channelized T1 ports
2 Packet over SONET interfaces
2 Channelized T3 ports
32768K bytes of non-volatile configuration memory.
4194304K bytes of physical memory.
921599K bytes of eUSB flash at bootflash:.
39004543K bytes of SATA hard disk at harddisk:.

Configuration register is 0x2102
```

Copying the Consolidated Package from the TFTP Server to the Router

The following example shows how to copy the consolidated package from the TFTP server to the router:

```
Router# dir bootflash:
Directory of bootflash:/

   11  drwx       16384   Dec 4 2007 04:32:46 -08:00  lost+found
86401  drwx        4096   Dec 4 2007 06:06:24 -08:00  .ssh
14401  drwx        4096   Dec 4 2007 06:06:36 -08:00  .rollback_timer
28801  drwx        4096  Mar 18 2008 17:31:17 -07:00  .prst_sync
43201  drwx        4096   Dec 4 2007 04:34:45 -08:00  .installer
   13  -rw-       45977   Apr 9 2008 16:48:46 -07:00  target_support_output.tgz.tgz

928862208 bytes total (712273920 bytes free)
Router# copy tftp bootflash:
Address or name of remote host []? 172.17.16.81
Source filename []?
/auto/tftp-users/user/asr1000rp1-adventerprisek9.02.01.00.122-33.XNA.bin
Destination filename [asr1000rp1-adventerprisek9.02.01.00.122-33.XNA.bin]?
Accessing
tftp://172.17.16.81//auto/tftp-users/user/asr1000rp1-adventerprisek9.02.01.00.122-33.XNA.b
in...
Loading /auto/tftp-users/user/asr1000rp1-adventerprisek9.02.01.00.122-33.XNA.bin from
172.17.16.81 (via GigabitEthernet0):
!!!
```

```
   11  drwx       16384   Dec 4 2007 04:32:46 -08:00  lost+found
86401  drwx        4096   Dec 4 2007 06:06:24 -08:00  .ssh
14401  drwx        4096   Dec 4 2007 06:06:36 -08:00  .rollback_timer
28801  drwx        4096  Mar 18 2008 17:31:17 -07:00  .prst_sync
43201  drwx        4096   Dec 4 2007 04:34:45 -08:00  .installer
   12  -rw-   208904396  May 28 2008 16:17:34 -07:00
asr1000rp1-adventerprisek9.02.01.00.122-33.XNA.bin
   13  -rw-       45977   Apr 9 2008 16:48:46 -07:00  target_support_output.tgz.tgz

928862208 bytes total (503156736 bytes free)
```

Configuring the Router to Boot Using the Consolidated Package Stored on the Router

The following example shows hot to configure the router to boot using the consolidated package stored on the router:

```
Router# dir bootflash:
Directory of bootflash:/

   11  drwx       16384   Dec 4 2007 04:32:46 -08:00  lost+found
86401  drwx        4096   Dec 4 2007 06:06:24 -08:00  .ssh
14401  drwx        4096   Dec 4 2007 06:06:36 -08:00  .rollback_timer
28801  drwx        4096  Mar 18 2008 17:31:17 -07:00  .prst_sync
43201  drwx        4096   Dec 4 2007 04:34:45 -08:00  .installer
   12  -rw-   208904396  May 28 2008 16:17:34 -07:00
asr1000rp1-adventerprisek9.02.01.00.122-33.XNA.bin
   13  -rw-       45977   Apr 9 2008 16:48:46 -07:00  target_support_output.tgz.tgz

928862208 bytes total (503156736 bytes free)

Router# config t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#boot system bootflash:asr1000rp1-adventerprisek9.02.01.00.122-33.XNA.bin
Router(config)#config-reg 0x2102
Router(config)#exit
Router#show run | include boot
boot-start-marker
boot system bootflash:asr1000rp1-adventerprisek9.02.01.00.122-33.XNA.bin
boot-end-marker
Router# copy run start
Destination filename [startup-config]?
Building configuration...
[OK]
Router# reload
<output removed for brevity>
User Access Verification

Username: user
Password:

Router>en
Password:
Router#show version
Cisco IOS Software, IOS-XE Software (PPC_LINUX_IOSD-ADVENTERPRISEK9-M), Version
12.2(33)XNA, RELEASE SOFTWARE (fc1)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2008 by Cisco Systems, Inc.
```

```
Compiled Thu 01-May-08 00:29 by mcpre


Cisco IOS-XE software, Copyright (c) 1986-2008 by Cisco Systems, Inc.
All rights reserved.  Certain components of Cisco IOS-XE software are
licensed under the GNU General Public License ("GPL") Version 2.0.  The
software code licensed under GPL Version 2.0 is free software that comes
with ABSOLUTELY NO WARRANTY.  You can redistribute and/or modify such
GPL code under the terms of GPL Version 2.0.  For more details, see the
documentation or "License Notice" file accompanying the IOS-XE software,
or the applicable URL provided on the flyer accompanying the IOS-XE
software.


ROM: IOS-XE ROMMON

 Router uptime is 3 minutes
Uptime for this control processor is 5 minutes
System returned to ROM by reload at 16:20:55 DST Wed May 28 2008
System image file is "bootflash:asr1000rp1-adventerprisek9.02.01.00.122-33.XNA.bin"
Last reload reason: Reload command



This product contains cryptographic features and is subject to United
States and local country laws governing import, export, transfer and
use. Delivery of Cisco cryptographic products does not imply
third-party authority to import, export, distribute or use encryption.
Importers, exporters, distributors and users are responsible for
compliance with U.S. and local country laws. By using this product you
agree to comply with applicable laws and regulations. If you are unable
to comply with U.S. and local laws, return this product immediately.

A summary of U.S. laws governing Cisco cryptographic products may be found at:
http://www.cisco.com/wwl/export/crypto/tool/stqrg.html

If you require further assistance please contact us by sending email to
export@cisco.com.

cisco ASR1006 (RP1) processor with 1772131K/6147K bytes of memory.
5 Gigabit Ethernet interfaces
8 Serial interfaces
8 Channelized T1 ports
2 Packet over SONET interfaces
2 Channelized T3 ports
32768K bytes of non-volatile configuration memory.
4194304K bytes of physical memory.
921599K bytes of eUSB flash at bootflash:.
39004543K bytes of SATA hard disk at harddisk:.

Configuration register is 0x2102
```

Extracting the SubPackages from a Consolidated Package into the Same File System

The following example shows how to extract the subpackages from a consolidated package into the same file system:

The asr1000rp1-adventerprisek9.02.01.00.122-33.XNA.bin consolidated package file is in bootflash, but no sub-packages are in the directory.

```
Router# dir bootflash:
Directory of bootflash:/

   11  drwx       16384   Dec 4 2007 04:32:46 -08:00  lost+found
86401  drwx        4096   Dec 4 2007 06:06:24 -08:00  .ssh
14401  drwx        4096   Dec 4 2007 06:06:36 -08:00  .rollback_timer
28801  drwx        4096  Mar 18 2008 17:31:17 -07:00  .prst_sync
43201  drwx        4096   Dec 4 2007 04:34:45 -08:00  .installer
   12  -rw-   208904396   May 9 2008 14:36:31 -07:00
asr1000rp1-adventerprisek9.02.01.00.122-33.XNA.bin
   13  -rw-       45977   Apr 9 2008 16:48:46 -07:00  target_support_output.tgz.tgz

928862208 bytes total (503156736 bytes free)

Router# request platform software package expand file
bootflash:asr1000rp1-adventerprisek9.02.01.00.122-33.XNA.bin
Verifying parameters
Validating package type
Copying package files

SUCCESS: Finished expanding all-in-one software package.
```

After entering the **request platform software package expand file bootflash:asr1000rp1-adventerprisek9.02.01.00.122-33.XNA.bin** command, noting that the **to** option is not used, the sub-packages are extracted from the consolidated package into bootflash:.

```
Router# dir bootflash:
Directory of bootflash:/

   11  drwx       16384   Dec 4 2007 04:32:46 -08:00  lost+found
86401  drwx        4096   Dec 4 2007 06:06:24 -08:00  .ssh
14401  drwx        4096   Dec 4 2007 06:06:36 -08:00  .rollback_timer
28801  drwx        4096  Mar 18 2008 17:31:17 -07:00  .prst_sync
43201  drwx        4096   Dec 4 2007 04:34:45 -08:00  .installer
   12  -rw-   208904396   May 9 2008 14:36:31 -07:00
asr1000rp1-adventerprisek9.02.01.00.122-33.XNA.bin
57611  -rw-    47071436  May 22 2008 11:26:23 -07:00
asr1000rp1-espbase.02.01.00.122-33.XNA.pkg
57602  -rw-        5740  May 22 2008 11:26:22 -07:00
asr1000rp1-packages-adventerprisek9.02.01.00.122-33.XNA.conf
57612  -rw-    20334796  May 22 2008 11:26:24 -07:00
asr1000rp1-rpaccess.02.01.00.122-33.XNA.pkg
57613  -rw-    22294732  May 22 2008 11:26:24 -07:00
asr1000rp1-rpbase.02.01.00.122-33.XNA.pkg
57614  -rw-    21946572  May 22 2008 11:26:25 -07:00
asr1000rp1-rpcontrol.02.01.00.122-33.XNA.pkg
57615  -rw-    48099532  May 22 2008 11:26:26 -07:00
asr1000rp1-rpios-adventerprisek9.02.01.00.122-33.XNA.pkg
57616  -rw-    34324684  May 22 2008 11:26:27 -07:00
asr1000rp1-sipbase.02.01.00.122-33.XNA.pkg
57617  -rw-    22124748  May 22 2008 11:26:28 -07:00
asr1000rp1-sipspa.02.01.00.122-33.XNA.pkg
57603  -rw-        6256  May 22 2008 11:26:28 -07:00  packages.conf
   13  -rw-       45977   Apr 9 2008 16:48:46 -07:00  target_support_output.tgz.tgz

928862208 bytes total (286662656 bytes free)
```

Extracting the SubPackages from a Consolidated Package into a Different File System

The following example shows how to extract the subpackages from a consolidated package into a different file system:

The asr1000rp1-adventerprisek9.02.01.00.122-33.XNA.bin consolidated package file is in usb0:

```
Router# dir usb0:
Directory of usb0:/

 1240  -rwx   208904396  May 27 2008 14:10:20 -07:00
asr1000rp1-adventerprisek9.02.01.00.122-33.XNA.bin

255497216 bytes total (40190464 bytes free)
```

No sub-packages are in the bootflash: directory.

```
Router# dir bootflash:
Directory of bootflash:/

   11  drwx       16384  Dec 13 2004 03:45:47 -08:00  lost+found
87937  drwx        4096  Dec 13 2004 03:46:45 -08:00  .rollback_timer
14657  drwx        4096  Dec 13 2004 03:47:17 -08:00  .installer
29313  drwx        4096  Dec 13 2004 03:53:00 -08:00  .ssh
   12  -rw-    33554432  Dec 13 2004 03:53:49 -08:00  nvram_00100
58625  drwx        4096   May 7 2008 17:27:51 -07:00  .prst_sync

945377280 bytes total (695246848 bytes free)

Router# request platform software package expand file
usb0:asr1000rp1-adventerprisek9.02.01.00.122-33.XNA.bin to bootflash:
Verifying parameters
Validating package type
Copying package files
SUCCESS: Finished expanding all-in-one software package.
```

After the **request platform software package expand file usb0:asr1000rp1-adventerprisek9.02.01.00.122-33.XNA.bin to bootflash:** command is entered, the sub-packages are now in the bootflash: directory.

```
Router# dir bootflash:
Directory of bootflash:/

   11  drwx       16384  Dec 13 2004 03:45:47 -08:00  lost+found
87937  drwx        4096  Dec 13 2004 03:46:45 -08:00  .rollback_timer
14657  drwx        4096  Dec 13 2004 03:47:17 -08:00  .installer
29313  drwx        4096  Dec 13 2004 03:53:00 -08:00  .ssh
   12  -rw-    33554432  Dec 13 2004 03:53:49 -08:00  nvram_00100
43979  -rw-    47071436  May 27 2008 14:25:01 -07:00
asr1000rp1-espbase.02.01.00.122-33.XNA.pkg
58625  drwx        4096   May 7 2008 17:27:51 -07:00  .prst_sync
43970  -rw-        5740  May 27 2008 14:25:00 -07:00
asr1000rp1-packages-adventerprisek9.02.01.00.122-33.XNA.conf
43980  -rw-    20334796  May 27 2008 14:25:01 -07:00
asr1000rp1-rpaccess.02.01.00.122-33.XNA.pkg
43981  -rw-    22294732  May 27 2008 14:25:02 -07:00
asr1000rp1-rpbase.02.01.00.122-33.XNA.pkg
```

```
43982  -rw-    21946572  May 27 2008 14:25:03 -07:00
asr1000rp1-rpcontrol.02.01.00.122-33.XNA.pkg
43983  -rw-    48099532  May 27 2008 14:25:04 -07:00
asr1000rp1-rpios-adventerprisek9.02.01.00.122-33.XNA.pkg
43984  -rw-    34324684  May 27 2008 14:25:05 -07:00
asr1000rp1-sipbase.02.01.00.122-33.XNA.pkg
43985  -rw-    22124748  May 27 2008 14:25:05 -07:00
asr1000rp1-sipspa.02.01.00.122-33.XNA.pkg
43971  -rw-        6256  May 27 2008 14:25:05 -07:00  packages.conf

945377280 bytes total (478752768 bytes free)
```

Configuring the Router to Boot Using the SubPackages

The following example shows how to configure the router to boot using the subpackages:

Enter the **dir bootflash:** command to confirm all sub-packages and the provisioning file are in the same file system:

```
Router# dir bootflash:
Directory of bootflash:/

   11  drwx       16384   Dec 4 2007 04:32:46 -08:00  lost+found
86401  drwx        4096   Dec 4 2007 06:06:24 -08:00  .ssh
14401  drwx        4096   Dec 4 2007 06:06:36 -08:00  .rollback_timer
28801  drwx        4096  Mar 18 2008 17:31:17 -07:00  .prst_sync
43201  drwx        4096   Dec 4 2007 04:34:45 -08:00  .installer
   12  -rw-   208904396   May 9 2008 14:36:31 -07:00
asr1000rp1-adventerprisek9.02.01.00.122-33.XNA.bin
57611  -rw-    47071436  May 22 2008 11:26:23 -07:00
asr1000rp1-espbase.02.01.00.122-33.XNA.pkg
57602  -rw-        5740  May 22 2008 11:26:22 -07:00
asr1000rp1-packages-adventerprisek9.02.01.00.122-33.XNA.conf
57612  -rw-    20334796  May 22 2008 11:26:24 -07:00
asr1000rp1-rpaccess.02.01.00.122-33.XNA.pkg
57613  -rw-    22294732  May 22 2008 11:26:24 -07:00
asr1000rp1-rpbase.02.01.00.122-33.XNA.pkg
57614  -rw-    21946572  May 22 2008 11:26:25 -07:00
asr1000rp1-rpcontrol.02.01.00.122-33.XNA.pkg
57615  -rw-    48099532  May 22 2008 11:26:26 -07:00
asr1000rp1-rpios-adventerprisek9.02.01.00.122-33.XNA.pkg
57616  -rw-    34324684  May 22 2008 11:26:27 -07:00
asr1000rp1-sipbase.02.01.00.122-33.XNA.pkg
57617  -rw-    22124748  May 22 2008 11:26:28 -07:00
asr1000rp1-sipspa.02.01.00.122-33.XNA.pkg
57603  -rw-        6256  May 22 2008 11:26:28 -07:00  packages.conf
   13  -rw-       45977   Apr 9 2008 16:48:46 -07:00  target_support_output.tgz.tgz

928862208 bytes total (286662656 bytes free)

Router# show running-config | include boot
boot-start-marker
boot-end-marker
Router# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# boot system bootflash:packages.conf
Router(config)# config-reg 0x2102
Router(config)# exit
Router# config t
*May 29 22:47:57.433: %SYS-5-CONFIG_I: Configured from console by user
Router# show running-config | include boot
```

```
boot-start-marker
boot system bootflash:packages.conf
boot-end-marker
Router# copy run start
Destination filename [startup-config]?
Building configuration...
[OK]
Router# reload
Proceed with reload? [confirm]

*May 29 22:51:54.194: %SYS-5-RELOAD: Reload requested by user on console. Reload Reason:
Reload command.


System Bootstrap, Version 12.2(33r)XN2, RELEASE SOFTWARE (fc1)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 2008 by cisco Systems, Inc.


Current image running: Boot ROM0
Last reset cause: LocalSoft

ASR1000-RP1 platform with 4194303 Kbytes of main memory

Located packages.conf
Image size 6256 inode num 57603, bks cnt 2 blk size 8*512
#
Located asr1000rp1-rpbase.02.01.00.122-33.XNA.pkg
Image size 22294732 inode num 57606, bks cnt 5444 blk size 8*512
################################################################################################
################################################################################################
####################################
Boot image size = 22294732 (0x15430cc) bytes

Using midplane macaddr
Package header rev 0 structure detected
Calculating SHA-1 hash...done
validate_package: SHA-1 hash:
        calculated 4ad33773:e1cb7492:db502416:4ad586f2:7c4d9701
        expected   4ad33773:e1cb7492:db502416:4ad586f2:7c4d9701
Image validated


PPC/IOS XE loader version: 0.0.3
loaded at:     00800000 01D45004
zimage at:     00807673 009B92D6
initrd at:     009BA000 01041CC9
isord at:      01042000 01D42800
avail ram:     00400000 00800000

Kernel load:
Uncompressing image... dst: 00000000 lim: 00400000 start: 00807673 size: 001B1C63...done.
Now booting the IOS XE kernel
%IOSXEBOOT-4-BOOT_SRC: (rp/0): Non-HD Boot
%IOSXEBOOT-4-BOOT_PARAMETER: (rp/0): Booting with custom BOOT_PARAM setting

              Restricted Rights Legend

Use, duplication, or disclosure by the Government is
subject to restrictions as set forth in subparagraph
(c) of the Commercial Computer Software - Restricted
Rights clause at FAR sec. 52.227-19 and subparagraph
(c) (1) (ii) of the Rights in Technical Data and Computer
Software clause at DFARS sec. 252.227-7013.
```

```
          cisco Systems, Inc.
          170 West Tasman Drive
          San Jose, California 95134-1706


Cisco IOS Software, IOS-XE Software (PPC_LINUX_IOSD-ADVENTERPRISEK9-M), Version
12.2(33)XNA, RELEASE SOFTWARE (fc1)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2008 by Cisco Systems, Inc.
Compiled Thu 01-May-08 00:29 by mcpre
Image text-base: 0x10064AF0, data-base: 0x137E0958


Cisco IOS-XE software, Copyright (c) 1986-2008 by Cisco Systems, Inc.
All rights reserved.  Certain components of Cisco IOS-XE software are
licensed under the GNU General Public License ("GPL") Version 2.0.  The
software code licensed under GPL Version 2.0 is free software that comes
with ABSOLUTELY NO WARRANTY.  You can redistribute and/or modify such
GPL code under the terms of GPL Version 2.0.  For more details, see the
documentation or "License Notice" file accompanying the IOS-XE software,
or the applicable URL provided on the flyer accompanying the IOS-XE
software.


This product contains cryptographic features and is subject to United
States and local country laws governing import, export, transfer and
use. Delivery of Cisco cryptographic products does not imply
third-party authority to import, export, distribute or use encryption.
Importers, exporters, distributors and users are responsible for
compliance with U.S. and local country laws. By using this product you
agree to comply with applicable laws and regulations. If you are unable
to comply with U.S. and local laws, return this product immediately.

A summary of U.S. laws governing Cisco cryptographic products may be found at:
http://www.cisco.com/wwl/export/crypto/tool/stqrg.html

If you require further assistance please contact us by sending email to
export@cisco.com.

cisco ASR1006 (RP1) processor with 1776227K/6147K bytes of memory.
5 Gigabit Ethernet interfaces
2 Packet over SONET interfaces
2 Channelized T3 ports
32768K bytes of non-volatile configuration memory.
4194304K bytes of physical memory.
921599K bytes of eUSB flash at bootflash:.
39004543K bytes of SATA hard disk at harddisk:.
Duplex is configured.Remove duplex configuration before enabling auto-negotiation


Press RETURN to get started!
<some output removed for brevity?
User Access Verification

Username: user
Password:

Router> enable
Password:
```

```
Router# show version
Cisco IOS Software, IOS-XE Software (PPC_LINUX_IOSD-ADVENTERPRISEK9-M), Version
12.2(33)XNA, RELEASE SOFTWARE (fc1)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2008 by Cisco Systems, Inc.
Compiled Thu 01-May-08 00:29 by mcpre


Cisco IOS-XE software, Copyright (c) 1986-2008 by Cisco Systems, Inc.
All rights reserved.  Certain components of Cisco IOS-XE software are
licensed under the GNU General Public License ("GPL") Version 2.0.  The
software code licensed under GPL Version 2.0 is free software that comes
with ABSOLUTELY NO WARRANTY.  You can redistribute and/or modify such
GPL code under the terms of GPL Version 2.0.  For more details, see the
documentation or "License Notice" file accompanying the IOS-XE software,
or the applicable URL provided on the flyer accompanying the IOS-XE
software.


ROM: IOS-XE ROMMON

 Router uptime is 1 minute
Uptime for this control processor is 3 minutes
System returned to ROM by reload at 15:51:54 DST Thu May 29 2008
System image file is "bootflash:packages.conf"
Last reload reason: Reload command



This product contains cryptographic features and is subject to United
States and local country laws governing import, export, transfer and
use. Delivery of Cisco cryptographic products does not imply
third-party authority to import, export, distribute or use encryption.
Importers, exporters, distributors and users are responsible for
compliance with U.S. and local country laws. By using this product you
agree to comply with applicable laws and regulations. If you are unable
to comply with U.S. and local laws, return this product immediately.

A summary of U.S. laws governing Cisco cryptographic products may be found at:
http://www.cisco.com/wwl/export/crypto/tool/stqrg.html

If you require further assistance please contact us by sending email to
export@cisco.com.

cisco ASR1006 (RP1) processor with 1776227K/6147K bytes of memory.
5 Gigabit Ethernet interfaces
8 Serial interfaces
8 Channelized T1 ports
2 Packet over SONET interfaces
2 Channelized T3 ports
32768K bytes of non-volatile configuration memory.
4194304K bytes of physical memory.
921599K bytes of eUSB flash at bootflash:.
39004543K bytes of SATA hard disk at harddisk:.

Configuration register is 0x2102
```

Backing Up Configuration Files

This section provides the following examples:

- Copying a Startup Configuration File to Bootflash, page B-14
- Copying a Startup Configuration File to an USB Flash Disk, page B-14
- Copying a Startup Configuration File to a TFTP Server, page B-15

Copying a Startup Configuration File to Bootflash

```
Router# dir bootflash:
Directory of bootflash:/

   11  drwx       16384   Dec 4 2007 04:32:46 -08:00  lost+found
86401  drwx        4096   Dec 4 2007 06:06:24 -08:00  .ssh
14401  drwx        4096   Dec 4 2007 06:06:36 -08:00  .rollback_timer
28801  drwx        4096  May 29 2008 16:31:41 -07:00  .prst_sync
43201  drwx        4096   Dec 4 2007 04:34:45 -08:00  .installer
   12  -rw-   208904396  May 28 2008 16:17:34 -07:00
asr1000rp1-adventerprisek9.02.01.00.122-33.XNA.bin

Router# copy nvram:startup-config bootflash:
Destination filename [startup-config]?

3517 bytes copied in 0.647 secs (5436 bytes/sec)

Router# dir bootflash:
Directory of bootflash:/

   11  drwx       16384   Dec 4 2007 04:32:46 -08:00  lost+found
86401  drwx        4096   Dec 4 2007 06:06:24 -08:00  .ssh
14401  drwx        4096   Dec 4 2007 06:06:36 -08:00  .rollback_timer
28801  drwx        4096  May 29 2008 16:31:41 -07:00  .prst_sync
43201  drwx        4096   Dec 4 2007 04:34:45 -08:00  .installer
   12  -rw-   208904396  May 28 2008 16:17:34 -07:00
asr1000rp1-adventerprisek9.02.01.00.122-33.XNA.bin
13 -rw-        7516   Jul 2 2008 15:01:39 -07:00  startup-config
```

Copying a Startup Configuration File to an USB Flash Disk

```
Router# dir usb0:
Directory of usb0:/

43261  -rwx   208904396  May 27 2008 14:10:20 -07:00
asr1000rp1-adventerprisek9.02.01.00.122-33.XNA.bin

255497216 bytes total (40190464 bytes free)

Router# copy nvram:startup-config usb0:
Destination filename [startup-config]?

3172 bytes copied in 0.214 secs (14822 bytes/sec)

Router# dir usb0:
Directory of usb0:/
```

```
43261  -rwx   208904396  May 27 2008 14:10:20 -07:00
asr1000rp1-adventerprisek9.02.01.00.122-33.XNA.bin
43262 -rwx        3172   Jul 2 2008 15:40:45 -07:00  startup-config

255497216 bytes total (40186880 bytes free)
```

Copying a Startup Configuration File to a TFTP Server

```
Router# copy bootflash:startup-config tftp:
Address or name of remote host []? 172.17.16.81
Destination filename [pe24_asr-1002-confg]? /auto/tftp-users/user/startup-config
!!
3517 bytes copied in 0.122 secs (28828 bytes/sec)
```

Enabling a Second IOS Process on a Single RP Using SSO

The **show platform** command output does not indicate a second IOS process is active:

```
Router# show platform
Chassis type: ASR1004

Slot      Type                State                 Insert time (ago)
--------- ------------------- --------------------- -----------------
0         ASR1000-SIP10       ok                    00:04:39
 0/0      SPA-5X1GE-V2        ok                    00:03:23
 0/1      SPA-2XT3/E3         ok                    00:03:18
R0        ASR1000-RP1         ok, active            00:04:39
F0        ASR1000-ESP10       ok, active            00:04:39
P0        ASR1004-PWR-AC      ok                    00:03:52
P1        ASR1004-PWR-AC      ok                    00:03:52

Slot      CPLD Version        Firmware Version
--------- ------------------- ----------------------------------------
0         07091401            12.2(33r)XN2
R0        07062111            12.2(33r)XN2
F0        07051680            12.2(33r)XN2
```

The **show redundancy states** command output shows that the redundancy state is nonredundant:

```
Router# show redundancy states
       my state = 13 -ACTIVE
     peer state = 1  -DISABLED
           Mode = Simplex
        Unit ID = 48

Redundancy Mode (Operational) = Non-redundant
Redundancy Mode (Configured)  = Non-redundant
Redundancy State              = Non Redundant
     Maintenance Mode = Disabled
    Manual Swact = disabled (system is simplex (no peer unit))
 Communications = Down      Reason: Simplex mode

   client count = 66
 client_notification_TMR = 30000 milliseconds
           RF debug mask = 0x0
```

SSO is configured:

```
Router# configure terminal
```

```
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# redundancy
Router(config-red)# mode sso
*May 27 19:43:43.539: %CMRP-6-DUAL_IOS_REBOOT_REQUIRED: R0/0: cmand:  Configuration must
be saved and the chassis must be rebooted for IOS redundancy changes to take effect
Router(config-red)# exit
Router(config)# exit
Router#show
*May 27 19:44:04.173: %SYS-5-CONFIG_I: Configured from console by user on console
```

The **show redundancy states** command output now indicates that SSO is configured but not operational:

```
Router# show redundancy states
       my state = 13 -ACTIVE
     peer state = 1  -DISABLED
           Mode = Simplex
         Unit ID = 48

Redundancy Mode (Operational) = Non-redundant
Redundancy Mode (Configured)  = sso
Redundancy State              = Non Redundant
     Maintenance Mode = Disabled
    Manual Swact = disabled (system is simplex (no peer unit))
 Communications = Down      Reason: Simplex mode

   client count = 66
 client_notification_TMR = 30000 milliseconds
           RF debug mask = 0x0
```

The **show platform** command output continues to show a second IOS process is still not running:

```
Router# show platform
Chassis type: ASR1004

Slot      Type                State                 Insert time (ago)
--------- ------------------- --------------------- -----------------
0         ASR1000-SIP10       ok                    00:05:53
 0/0      SPA-5X1GE-V2        ok                    00:04:37
 0/1      SPA-2XT3/E3         ok                    00:04:32
R0        ASR1000-RP1         ok, active            00:05:53
F0        ASR1000-ESP10       ok, active            00:05:53
P0        ASR1004-PWR-AC      ok                    00:05:06
P1        ASR1004-PWR-AC      ok                    00:05:06

Slot      CPLD Version        Firmware Version
--------- ------------------- ----------------------------------------
0         07091401            12.2(33r)XN2
R0        07062111            12.2(33r)XN2
F0        07051680            12.2(33r)XN2

Router# copy running-config startup-config
Destination filename [startup-config]?
Building configuration...
[OK]
```

The router needs to be reloaded after saving the running configuration to make SSO the operational redundancy mode:

```
Router# reload
Proceed with reload? [confirm]

*May 27 19:45:16.917: %SYS-5-RELOAD: Reload requested by user on console. Reload Reason:
Reload command.
```

```
System Bootstrap, Version 12.2(33r)XN2, RELEASE SOFTWARE (fc1)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 2008 by cisco Systems, Inc.


Current image running: Boot ROM1
Last reset cause: LocalSoft

ASR1000-RP1 platform with 4194303 Kbytes of main memory

Located packages.conf
Image size 6256 inode num 43971, bks cnt 2 blk size 8*512
#
Located asr1000rp1-rpbase.02.01.00.122-33.XNA.pkg
Image size 22294732 inode num 43974, bks cnt 5444 blk size 8*512
################################################################################################
################################################################################################
####################################
Boot image size = 22294732 (0x15430cc) bytes

Using midplane macaddr
Package header rev 0 structure detected
Calculating SHA-1 hash...done
validate_package: SHA-1 hash:
        calculated 4ad33773:e1cb7492:db502416:4ad586f2:7c4d9701
        expected   4ad33773:e1cb7492:db502416:4ad586f2:7c4d9701
Image validated


PPC/IOS XE loader version: 0.0.3
loaded at:     00800000 01D45004
zimage at:     00807673 009B92D6
initrd at:     009BA000 01041CC9
isord at:      01042000 01D42800
avail ram:     00400000 00800000

Kernel load:
Uncompressing image... dst: 00000000 lim: 00400000 start: 00807673 size: 001B1C63...done.
Now booting the IOS XE kernel
%IOSXEBOOT-4-BOOT_SRC: (rp/0): Non-HD Boot
%IOSXEBOOT-4-BOOT_PARAMETER: (rp/0): Booting with custom BOOT_PARAM setting

              Restricted Rights Legend

Use, duplication, or disclosure by the Government is
subject to restrictions as set forth in subparagraph
(c) of the Commercial Computer Software - Restricted
Rights clause at FAR sec. 52.227-19 and subparagraph
(c) (1) (ii) of the Rights in Technical Data and Computer
Software clause at DFARS sec. 252.227-7013.

           cisco Systems, Inc.
           170 West Tasman Drive
           San Jose, California 95134-1706



Cisco IOS Software, IOS-XE Software (PPC_LINUX_IOSD-ADVENTERPRISEK9-M), Version
12.2(33)XNA, RELEASE SOFTWARE (fc1)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2008 by Cisco Systems, Inc.
Compiled Thu 01-May-08 00:29 by mcpre
```

```
Image text-base: 0x10064AF0, data-base: 0x137E0958



Cisco IOS-XE software, Copyright (c) 1986-2008 by Cisco Systems, Inc.
All rights reserved.  Certain components of Cisco IOS-XE software are
licensed under the GNU General Public License ("GPL") Version 2.0.  The
software code licensed under GPL Version 2.0 is free software that comes
with ABSOLUTELY NO WARRANTY.  You can redistribute and/or modify such
GPL code under the terms of GPL Version 2.0.  For more details, see the
documentation or "License Notice" file accompanying the IOS-XE software,
or the applicable URL provided on the flyer accompanying the IOS-XE
software.



This product contains cryptographic features and is subject to United
States and local country laws governing import, export, transfer and
use. Delivery of Cisco cryptographic products does not imply
third-party authority to import, export, distribute or use encryption.
Importers, exporters, distributors and users are responsible for
compliance with U.S. and local country laws. By using this product you
agree to comply with applicable laws and regulations. If you are unable
to comply with U.S. and local laws, return this product immediately.

A summary of U.S. laws governing Cisco cryptographic products may be found at:
http://www.cisco.com/wwl/export/crypto/tool/stqrg.html

If you require further assistance please contact us by sending email to
export@cisco.com.

cisco ASR1004 (RP1) processor with 752227K/6147K bytes of memory.
5 Gigabit Ethernet interfaces
32768K bytes of non-volatile configuration memory.
4194304K bytes of physical memory.
937983K bytes of eUSB flash at bootflash:.
39004543K bytes of SATA hard disk at harddisk:.
253424K bytes of USB flash at usb0:.



Press RETURN to get started!

<some output omitted for brevity>

User Access Verification

Username: user
Password:

Router> enable
Password:
```

The **show platform** command output now indicates a second IOS process is active on RP 0:

```
Router# show platform
Chassis type: ASR1004

Slot      Type                State                 Insert time (ago)
--------- ------------------- --------------------- -----------------
0         ASR1000-SIP10       ok                    00:29:34
 0/0      SPA-5X1GE-V2        ok                    00:28:13
 0/1      SPA-2XT3/E3         ok                    00:28:18
```

```
R0         ASR1000-RP1          ok                    00:29:34
 R0/0                           ok, active            00:29:34
 R0/1                           ok, standby           00:27:49
F0         ASR1000-ESP10        ok, active            00:29:34
P0         ASR1004-PWR-AC       ok                    00:28:47
P1         ASR1004-PWR-AC       ok                    00:28:47

Slot       CPLD Version         Firmware Version
---------- -------------------- ----------------------------------------
0          07091401             12.2(33r)XN2
R0         07062111             12.2(33r)XN2
F0         07051680             12.2(33r)XN2
```

The **show redundancy states** command shows SSO as the operational redundancy mode:

```
Router# show redundancy states
       my state = 13 -ACTIVE
     peer state = 8  -STANDBY HOT
           Mode = Duplex
        Unit ID = 48

Redundancy Mode (Operational) = sso
Redundancy Mode (Configured)  = sso
Redundancy State              = sso
     Maintenance Mode = Disabled
    Manual Swact = enabled
 Communications = Up

   client count = 66
 client_notification_TMR = 30000 milliseconds
           RF debug mask = 0x0
```

ISSU—Consolidated Package Upgrade

In the following example, ISSU is used to upgrade two RPs in the same Cisco ASR 1000 Series Router to consolidated packages. The RPs are being upgraded from Cisco IOS XE Release 2.1.0, which uses Cisco IOS Release 12.2(33)XNA, to Cisco IOS XE Release 2.1.1, which uses Cisco IOS Release 12.2(33)XNA1.

The example starts with the user logged into RP 0, the active RP.

```
Router# show version
Cisco IOS Software, IOS-XE Software (PPC_LINUX_IOSD-ADVENTERPRISEK9-M), Version
12.2(33)XNA, RELEASE SOFTWARE (fc1)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2008 by Cisco Systems, Inc.
Compiled Thu 01-May-08 00:29 by mcpre


Cisco IOS-XE software, Copyright (c) 1986-2008 by Cisco Systems, Inc.
All rights reserved.  Certain components of Cisco IOS-XE software are
licensed under the GNU General Public License ("GPL") Version 2.0.  The
software code licensed under GPL Version 2.0 is free software that comes
with ABSOLUTELY NO WARRANTY.  You can redistribute and/or modify such
GPL code under the terms of GPL Version 2.0.  For more details, see the
documentation or "License Notice" file accompanying the IOS-XE software,
or the applicable URL provided on the flyer accompanying the IOS-XE
software.
```

```
ROM: IOS-XE ROMMON

 Router uptime is 2 weeks, 2 hours, 54 minutes
Uptime for this control processor is 1 hour, 7 minutes
System returned to ROM by reload at 15:24:15 DST Mon Jul 21 2008
System image file is "bootflash:asr1000rp1-adventerprisek9.02.01.00.122-33.XNA.bin"
Last reload reason: redundancy force-switchover



This product contains cryptographic features and is subject to United
States and local country laws governing import, export, transfer and
use. Delivery of Cisco cryptographic products does not imply
third-party authority to import, export, distribute or use encryption.
Importers, exporters, distributors and users are responsible for
compliance with U.S. and local country laws. By using this product you
agree to comply with applicable laws and regulations. If you are unable
to comply with U.S. and local laws, return this product immediately.

A summary of U.S. laws governing Cisco cryptographic products may be found at:
http://www.cisco.com/wwl/export/crypto/tool/stqrg.html

If you require further assistance please contact us by sending email to
export@cisco.com.

cisco ASR1006 (RP1) processor with 1772131K/6147K bytes of memory.
8 FastEthernet interfaces
5 Gigabit Ethernet interfaces
8 Serial interfaces
8 Channelized T1 ports
2 Packet over SONET interfaces
2 Channelized T3 ports
32768K bytes of non-volatile configuration memory.
4194304K bytes of physical memory.
921599K bytes of eUSB flash at bootflash:.
39004543K bytes of SATA hard disk at harddisk:.

Configuration register is 0x2102

Router# dir bootflash:
Directory of bootflash:/

   11  drwx       16384   Dec 4 2007 04:32:46 -08:00  lost+found
86401  drwx        4096   Dec 4 2007 06:06:24 -08:00  .ssh
14401  drwx        4096   Dec 4 2007 06:06:36 -08:00  .rollback_timer
28801  drwx        4096  Jul 21 2008 15:29:25 -07:00  .prst_sync
43201  drwx        4096   Dec 4 2007 04:34:45 -08:00  .installer
   12  -rw-   208904396  May 28 2008 16:17:34 -07:00
asr1000rp1-adventerprisek9.02.01.00.122-33.XNA.bin
13 -rw-        7516   Jul 2 2008 15:01:39 -07:00  startup-config
   14 -rw-       45977   Apr 9 2008 16:48:46 -07:00  target_support_output.tgz.tgz

928862208 bytes total (76644352 bytes free)

Router# dir stby-bootflash:
Directory of stby-bootflash:/

   11  drwx       16384  Dec 13 2004 03:45:47 -08:00  lost+found
87937  drwx        4096  Jul 17 2008 16:43:34 -07:00  .rollback_timer
14657  drwx        4096  Jul 17 2008 16:43:34 -07:00  .installer
29313  drwx        4096  Dec 13 2004 03:53:00 -08:00  .ssh
   12  -rw-    33554432  Dec 13 2004 03:53:49 -08:00  nvram_00100
```

```
   13  -rw-   208904396   Jun 5 2008 20:12:53 -07:00
asr1000rp1-adventerprisek9.02.01.00.122-33.XNA.bin
58625  drwx        4096  Jul 21 2008 15:32:59 -07:00  .prst_sync

945377280 bytes total (276652032 bytes free)


Router# copy tftp bootflash:
Address or name of remote host []? 172.17.16.81
Source filename []?
/auto/tftp-users/user/asr1000rp1-adventerprisek9.02.01.01.122-33.XNA1.bin
Destination filename [asr1000rp1-adventerprisek9.02.01.01.122-33.XNA1.bin]?
Accessing
tftp://172.17.16.81//auto/tftp-users/user/asr1000rp1-adventerprisek9.02.01.01.122-33.XNA1.
bin...
Loading /auto/tftp-users/user/asr1000rp1-adventerprisek9.02.01.01.122-33.XNA1.bin from
172.17.16.81 (via GigabitEthernet0):
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!
[OK - 209227980 bytes]

209227980 bytes copied in 329.215 secs (635536 bytes/sec)

Router# copy bootflash:asr1000rp1-adventerprisek9.02.01.01.122-33.XNA1.bin stby-bootflash:
Destination filename [asr1000rp1-adventerprisek9.02.01.01.122-33.XNA1.bin]?
Copy in progress...CCCCCCCC
<output removed for brevity>
209227980 bytes copied in 434.790 secs (481216 bytes/sec)

Router# dir bootflash:
Directory of bootflash:/

   11  drwx       16384   Dec 4 2007 04:32:46 -08:00  lost+found
86401  drwx        4096   Dec 4 2007 06:06:24 -08:00  .ssh
14401  drwx        4096   Dec 4 2007 06:06:36 -08:00  .rollback_timer
28801  drwx        4096  Jul 21 2008 15:29:25 -07:00  .prst_sync
43201  drwx        4096   Dec 4 2007 04:34:45 -08:00  .installer
   12  -rw-   208904396  May 28 2008 16:17:34 -07:00
asr1000rp1-adventerprisek9.02.01.00.122-33.XNA.bin
   15  -rw-        7516   Jul 2 2008 15:01:39 -07:00  startup-config
   13  -rw-       45977   Apr 9 2008 16:48:46 -07:00  target_support_output.tgz.tgz
   16  -rw-   209227980  Jul 17 2008 16:06:58 -07:00
asr1000rp1-adventerprisek9.02.01.01.122-33.XNA1.bin

928862208 bytes total (76644352 bytes free)
Router# dir stby-bootflash:
Directory of stby-bootflash:/

   11  drwx       16384  Dec 13 2004 03:45:47 -08:00  lost+found
87937  drwx        4096  Jul 17 2008 16:43:34 -07:00  .rollback_timer
14657  drwx        4096  Jul 17 2008 16:43:34 -07:00  .installer
29313  drwx        4096  Dec 13 2004 03:53:00 -08:00  .ssh
   12  -rw-    33554432  Dec 13 2004 03:53:49 -08:00  nvram_00100
   13  -rw-   208904396   Jun 5 2008 20:12:53 -07:00
asr1000rp1-adventerprisek9.02.01.00.122-33.XNA.bin
58625  drwx        4096  Jul 21 2008 15:32:59 -07:00  .prst_sync
```

```
   14  -rw-   209227980  Jul 17 2008 16:16:07 -07:00
asr1000rp1-adventerprisek9.02.01.01.122-33.XNA1.bin

945377280 bytes total (276652032 bytes free)

Router# issu loadversion rp 1 file
stby-bootflash:asr1000rp1-adventerprisek9.02.01.01.122-33.XNA1.bin
--- Starting installation state synchronization ---
Finished installation state synchronization

--- Starting file path checking ---
Finished file path checking

--- Starting system installation readiness checking ---
Finished system installation readiness checking

--- Starting installation changes ---
Setting up image to boot on next reset
Starting automatic rollback timer
Finished installation changes

SUCCESS: Software will now load.

Router#
*Jul 21 23:34:27.206: %ASR1000_OIR-6-OFFLINECARD: Card (rp) offline in slot R1
*Jul 21 23:34:27.271: %REDUNDANCY-3-STANDBY_LOST: Standby processor fault
(PEER_NOT_PRESENT)
*Jul 21 23:34:27.271: %REDUNDANCY-3-STANDBY_LOST: Standby processor fault (PEER_DOWN)
*Jul 21 23:34:27.271: %REDUNDANCY-3-STANDBY_LOST: Standby processor fault
(PEER_REDUNDANCY_STATE_CHANGE)
*Jul 21 23:37:05.528: %ASR1000_OIR-6-ONLINECARD: Card (rp) online in slot R1
*Jul 21 23:37:25.480: %REDUNDANCY-5-PEER_MONITOR_EVENT: Active detected a standby
insertion (raw-event=PEER_FOUND(4))

*Jul 21 23:37:25.480: %REDUNDANCY-5-PEER_MONITOR_EVENT: Active detected a standby
insertion (raw-event=PEER_REDUNDANCY_STATE_CHANGE(5))
Finished installation state synchronization


*Jul 21 23:37:26.349: %REDUNDANCY-3-IPC: IOS versions do not match.
*Jul 21 23:38:47.172: %HA_CONFIG_SYNC-6-BULK_CFGSYNC_SUCCEED: Bulk Sync succeeded
*Jul 21 23:38:47.173: %RF-5-RF_TERMINAL_STATE: Terminal state reached for (SSO)

Router# issu runversion
--- Starting installation state synchronization ---
Finished installation state synchronization

Initiating active RP failover
SUCCESS: Standby RP will now become active

Router#


System Bootstrap, Version 12.2(33r)XN2, RELEASE SOFTWARE (fc1)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 2008 by cisco Systems, Inc.

<additional output removed for brevity>

*Jul 21 23:43:31.970: %SYS-5-RESTART: System restarted --
Cisco IOS Software, IOS-XE Software (PPC_LINUX_IOSD-ADVENTERPRISEK9-M), Version
12.2(33)XNA, RELEASE SOFTWARE (fc1)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2008 by Cisco Systems, Inc.
```

```
Compiled Thu 01-May-08 00:29 by mcpre
*Jul 21 23:43:31.978: %SSH-5-ENABLED: SSH 1.99 has been enabled
*Jul 21 23:43:35.196: Relay: standby progression done
*Jul 21 23:43:35.197: %PLATFORM-6-RF_PROG_SUCCESS: RF state STANDBY HOT
```

At this point of the process, the user has to log onto RP1, which started the upgrade as the standby RP but is the active RP after the switchover.

The following commands are entered from RP1:

```
Router# issu acceptversion
Cancelling rollback timer
SUCCESS: Rollback timer cancelled

Router# issu commitversion
--- Starting installation changes ---
Cancelling rollback timer
Saving image changes
Finished installation changes


Building configuration...
[OK]
SUCCESS: version committed: bootflash:asr1000rp1-adventerprisek9.02.01.01.122-33.XNA1.bin

Router# show version
Cisco IOS Software, IOS-XE Software (PPC_LINUX_IOSD-ADVENTERPRISEK9-M), Version
12.2(33)XNA1, RELEASE SOFTWARE (fc1)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2008 by Cisco Systems, Inc.
Compiled Tue 08-Jul-08 14:40 by mcpre


Cisco IOS-XE software, Copyright (c) 1986-2008 by Cisco Systems, Inc.
All rights reserved.  Certain components of Cisco IOS-XE software are
licensed under the GNU General Public License ("GPL") Version 2.0.  The
software code licensed under GPL Version 2.0 is free software that comes
with ABSOLUTELY NO WARRANTY.  You can redistribute and/or modify such
GPL code under the terms of GPL Version 2.0.  For more details, see the
documentation or "License Notice" file accompanying the IOS-XE software,
or the applicable URL provided on the flyer accompanying the IOS-XE
software.


ROM: IOS-XE ROMMON

 Router uptime is 2 weeks, 3 hours, 8 minutes
Uptime for this control processor is 11 minutes
System returned to ROM by reload at 15:29:24 DST Mon Jul 21 2008
System image file is "bootflash:asr1000rp1-adventerprisek9.02.01.01.122-33.XNA1.bin"
Last reload reason: EHSA standby down



This product contains cryptographic features and is subject to United
States and local country laws governing import, export, transfer and
use. Delivery of Cisco cryptographic products does not imply
third-party authority to import, export, distribute or use encryption.
Importers, exporters, distributors and users are responsible for
compliance with U.S. and local country laws. By using this product you
agree to comply with applicable laws and regulations. If you are unable
to comply with U.S. and local laws, return this product immediately.
```

```
A summary of U.S. laws governing Cisco cryptographic products may be found at:
http://www.cisco.com/wwl/export/crypto/tool/stqrg.html

If you require further assistance please contact us by sending email to
export@cisco.com.

cisco ASR1006 (RP1) processor with 1772046K/6147K bytes of memory.
8 FastEthernet interfaces
5 Gigabit Ethernet interfaces
8 Serial interfaces
8 Channelized T1 ports
2 Packet over SONET interfaces
2 Channelized T3 ports
32768K bytes of non-volatile configuration memory.
4194304K bytes of physical memory.
937983K bytes of eUSB flash at bootflash:.
39004543K bytes of SATA hard disk at harddisk:.
253424K bytes of USB flash at usb0:.

Configuration register is 0x2102

Router# show running-config | include boot
boot-start-marker
boot system bootflash:asr1000rp1-adventerprisek9.02.01.01.122-33.XNA1.bin
boot system bootflash:asr1000rp1-adventerprisek9.02.01.00.122-33.XNA.bin
boot-end-marker
Router#
```

At this point of the procedure, the upgrade of RP1 is complete.

The following procedure uses the same steps to upgrade RP0. The user is still on RP1 when this process begins:

```
Router# show platform
Chassis type: ASR1006

Slot      Type                State                 Insert time (ago)
--------- ------------------- --------------------- -----------------
0         ASR1000-SIP10       ok                    00:16:10
 0/0      SPA-5X1GE-V2        ok                    00:10:50
 0/1      SPA-8X1FE-TX-V2     ok                    00:10:46
 0/2      SPA-2XCT3/DS0       ok                    00:10:42
1         ASR1000-SIP10       ok                    00:16:10
 1/0      SPA-2XOC3-POS       ok                    00:10:49
 1/1      SPA-8XCHT1/E1       ok                    00:10:45
 1/2      SPA-2XT3/E3         ok                    00:10:41
R0        ASR1000-RP1         ok, standby           00:16:10
R1        ASR1000-RP1         ok, active            00:16:10
F0        ASR1000-ESP10       ok, standby           00:16:10
F1        ASR1000-ESP10       ok, active            00:16:10
P0        ASR1006-PWR-AC      ok                    00:15:07
P1        ASR1006-FAN         ok                    00:15:07

Slot      CPLD Version        Firmware Version
--------- ------------------- ----------------------------------------
0         06120701            12.2(33r)XN2
1         06120701            12.2(33r)XN2
R0        07082312            12.2(33r)XN2
R1        07062111            12.2(33r)XN2
F0        07051680            12.2(33r)XN2
F1        07051680            12.2(33r)XN2
```

```
Router# issu loadversion rp 0 file
stby-bootflash:asr1000rp1-adventerprisek9.02.01.01.122-33.XNA1.bin
--- Starting installation state synchronization ---
Finished installation state synchronization

--- Starting file path checking ---
Finished file path checking

--- Starting system installation readiness checking ---
Finished system installation readiness checking

--- Starting installation changes ---
Setting up image to boot on next reset
Starting automatic rollback timer
Finished installation changes

SUCCESS: Software will now load.

Router#
*Jul 21 23:53:41.218: %ASR1000_OIR-6-OFFLINECARD: Card (rp) offline in slot R0
*Jul 21 23:53:41.256: %REDUNDANCY-3-STANDBY_LOST: Standby processor fault
(PEER_NOT_PRESENT)
*Jul 21 23:53:41.256: %REDUNDANCY-3-STANDBY_LOST: Standby processor fault (PEER_DOWN)
*Jul 21 23:53:41.256: %REDUNDANCY-3-STANDBY_LOST: Standby processor fault
(PEER_REDUNDANCY_STATE_CHANGE)
*Jul 21 23:53:42.423: %IP-4-DUPADDR: Duplicate address 172.29.52.155 on GigabitEthernet0,
sourced by 001a.3046.e3ff
*Jul 21 23:56:19.885: %ASR1000_OIR-6-ONLINECARD: Card (rp) online in slot R0
*Jul 21 23:56:39.324: %REDUNDANCY-5-PEER_MONITOR_EVENT: Active detected a standby
insertion (raw-event=PEER_FOUND(4))

*Jul 21 23:56:39.324: %REDUNDANCY-5-PEER_MONITOR_EVENT: Active detected a standby
insertion (raw-event=PEER_REDUNDANCY_STATE_CHANGE(5))

*Jul 21 23:58:03.660: %HA_CONFIG_SYNC-6-BULK_CFGSYNC_SUCCEED: Bulk Sync succeeded
*Jul 21 23:58:03.661: %RF-5-RF_TERMINAL_STATE: Terminal state reached for (SSO)

Router# issu runversion
--- Starting installation state synchronization ---
Finished installation state synchronization

Initiating active RP failover
SUCCESS: Standby RP will now become active

Router#


System Bootstrap, Version 12.2(33r)XN2, RELEASE SOFTWARE (fc1)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 2008 by cisco Systems, Inc.

<Output removed for brevity>
```

After the switchover, the user must log onto RP0.

The remaining commands are all entered from RP0:

```
Router# issu acceptversion
Cancelling rollback timer
SUCCESS: Rollback timer cancelled

Router# issu commitversion
```

```
*Jul 22 00:10:45.488: %HA_CONFIG_SYNC-6-BULK_CFGSYNC_SUCCEED: Bulk Sync succeeded
*Jul 22 00:10:45.489: %RF-5-RF_TERMINAL_STATE: Terminal state reached for (SSO)
--- Starting installation changes ---
Cancelling rollback timer
Saving image changes
Finished installation changes


Building configuration...
[OK]
SUCCESS: version committed: bootflash:asr1000rp1-adventerprisek9.02.01.01.122-33.XNA1.bin

Router# show version
Cisco IOS Software, IOS-XE Software (PPC_LINUX_IOSD-ADVENTERPRISEK9-M), Version
12.2(33)XNA1, RELEASE SOFTWARE (fc1)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2008 by Cisco Systems, Inc.
Compiled Tue 08-Jul-08 14:40 by mcpre


Cisco IOS-XE software, Copyright (c) 1986-2008 by Cisco Systems, Inc.
All rights reserved.  Certain components of Cisco IOS-XE software are
licensed under the GNU General Public License ("GPL") Version 2.0.  The
software code licensed under GPL Version 2.0 is free software that comes
with ABSOLUTELY NO WARRANTY.  You can redistribute and/or modify such
GPL code under the terms of GPL Version 2.0.  For more details, see the
documentation or "License Notice" file accompanying the IOS-XE software,
or the applicable URL provided on the flyer accompanying the IOS-XE
software.


ROM: IOS-XE ROMMON

 Router uptime is 2 weeks, 3 hours, 33 minutes
Uptime for this control processor is 16 minutes
System returned to ROM by reload at 15:24:15 DST Mon Jul 21 2008
System image file is "bootflash:asr1000rp1-adventerprisek9.02.01.01.122-33.XNA1.bin"
Last reload reason: EHSA standby down



This product contains cryptographic features and is subject to United
States and local country laws governing import, export, transfer and
use. Delivery of Cisco cryptographic products does not imply
third-party authority to import, export, distribute or use encryption.
Importers, exporters, distributors and users are responsible for
compliance with U.S. and local country laws. By using this product you
agree to comply with applicable laws and regulations. If you are unable
to comply with U.S. and local laws, return this product immediately.

A summary of U.S. laws governing Cisco cryptographic products may be found at:
http://www.cisco.com/wwl/export/crypto/tool/stqrg.html

If you require further assistance please contact us by sending email to
export@cisco.com.

cisco ASR1006 (RP1) processor with 1772046K/6147K bytes of memory.
8 FastEthernet interfaces
5 Gigabit Ethernet interfaces
8 Serial interfaces
8 Channelized T1 ports
2 Packet over SONET interfaces
2 Channelized T3 ports
32768K bytes of non-volatile configuration memory.
```

```
4194304K bytes of physical memory.
921599K bytes of eUSB flash at bootflash:.
39004543K bytes of SATA hard disk at harddisk:.

Configuration register is 0x2102

Router# show running-config | include boot
boot-start-marker
boot system bootflash:asr1000rp1-adventerprisek9.02.01.01.122-33.XNA1.bin
boot system bootflash:asr1000rp1-adventerprisek9.02.01.00.122-33.XNA.bin
boot-end-marker
Router#
```

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: www.cisco.com/go/trademarks. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1110R)

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.

Cisco ASR 1000 Series Aggregation Services Routers Software Configuration Guide

Copyright © 2008-2014 Cisco Systems, Inc. All rights reserved.